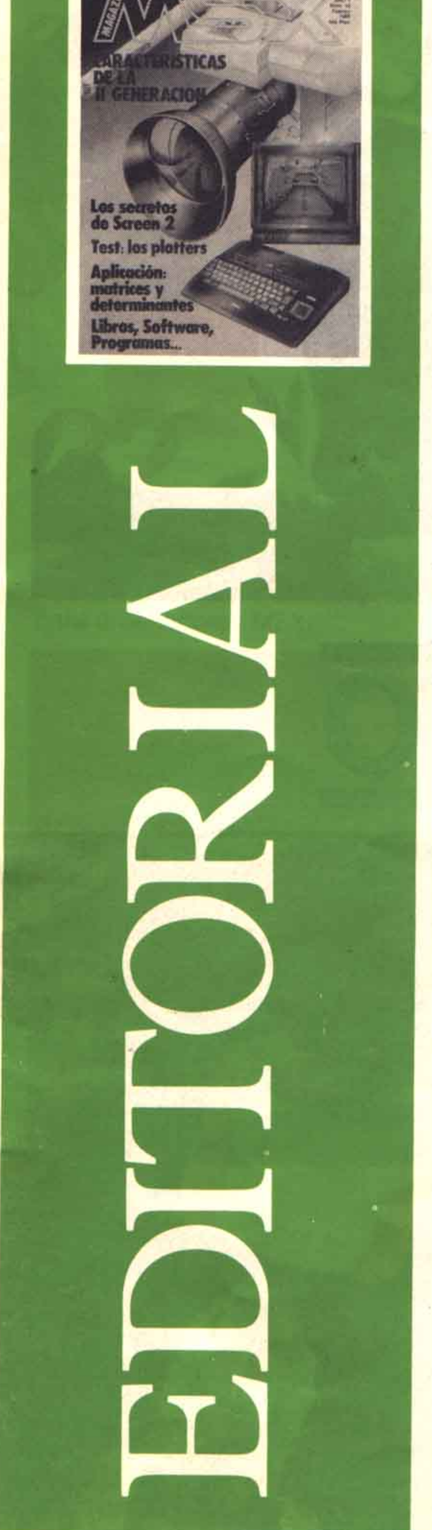


MAGAZINE MSX
AÑO I
Núm. 10
Febrero
1986
300 Ptas.
CARACTERISTICAS
DE LA
II GENERACION
Los secretos
de Screen 2
Test: los plotters
Aplicación:
matrices y
determinantes
Libros, Software,
Programas...





El mes pasado publicábamos una encuesta en la que pedíamos vuestra colaboración e ideas para mejorar en la medida de lo posible y entre todos, esta publicación dedicada al estándar MSX. La respuesta ha sido totalmente inesperada.

Diariamente llegan a nuestra redacción docenas de ellas que poco a poco, analizamos y aunque la tónica general es de agrado hacia la publicación, nos consta que deseáis ver ciertas secciones mejoradas, algunas ampliadas y otras,... mejor transformarlas.

Procuraremos agradar a todos para que de alguna forma os hagamos llegar un producto final bueno e interesante. Para ello, inicialmente, realizaremos más concursos y encuestas periódicas (cada tres o cuatro meses, con sorteos de libros, programas, etc.), así como pensamos crear alguna sección donde seáis vosotros los protagonistas.

Por otro lado, las noticias no podían ser mejores. A la piratería le ha surgido un serio contrincante; la tarjeta inteligente. Existen dos tipos distintos de tarjetas, cuya diferencia la encontraremos en el número de terminales así como en la capacidad de almacenamiento de cada una y en el tipo de grabación, ya que unas son ROM mientras las otras son EPROM. La comercialización de dichas tarjetas por Serma, aportará un nuevo soporte de información, cuyo tamaño —sumamente pequeño— permitirá un manejo fácil. Con una capacidad que rondará los 64K o 256K y un precio muy asequible, su futuro es muy prometedor.

Y hablando de futuro, podemos decir que, en unos meses veremos aparecer la 2.ª generación de ordenadores MSX, que estarán dispuestos a luchar por ese mercado abierto a los modelos de 128K, tan de moda en la actualidad, con ordenadores tan conocidos como el Commodore 128, Spectrum 128 o el Amstrad 6128. Sus armas son contundentes y, aunque su precio lo delate, estará orientado hacia el mercado semiprofesional.



SUMARIO











AGIE

NOTICIAS

Fallado el concurso de software Sony MSX

Recientemente ha sido fallado el concurso de software para MSX convocado el pasado verano por Sony España. Como se recordará las reglas admitían a concurso dos categorías: programas didácticos desarrollados por centros docentes y programas de tema libre desarrollados por usuarios y ordenadores MSX.

Fco. Javier Rodríguez Arévalo y Jorge Francés Rovira son los creadores de «PENIBAS», el programa ganador en la rama de los didácticos, consistente en una demostración visual de la evolución de la información dentro del ordenador, ya sean datos o programas, así como de la forma en que los periféricos más comunes se comunican con el ordenador. El premio consiste en 500.000 ptas. a repartir entre ambos y otras 500.000 ptas. para el Instituto de Formación Profesional El Vendrell, de Tarragona, como centro docente.

En cuanto a la categoría de tema libre, decir que estaba previsto entregar un primer premio de 500.000 ptas. y otros cinco de 100.000 ptas., pero debido al mérito y la calidad de dos programas, el jurado optó por entregar dos premios de 300.000 ptas. y cuatro de 100.000 ptas.

Los dos primeros premios han sido para Ricard Calvo i Catalán, por su programa «P.O.S.», y Javier Vila Robert, por "HAL". El primer programa permite visualizar a imprimir diferentes perspectivas de un dibujo previamente definido, mientras que el segundo funciona como co-intérprete del intérprete de Basic MSX residente. Dispone de más de veinte comandos y funciones que amplían las posibilidades del intérprete. Los cuatro premios de 100.000 ptas se han repartido así:

Uno para Jesús Asín Gascón, por su programa «DETECTIVE O.WELLES», que consiste en descubrir, en el interior de una mansión y por el método de exclusión, el lugar exacto del crimen, el arma homicida y el asesino. Otro al programa «DIGIT», que analiza mediante simulación los circuitos digitales de uso general, creado por José L. Ramos Suárez. Un tercero para «FLOP-CHOP», de Juan Roig Ferrán y, por último, uno para el programa didáctico «HALLEY» de Enrique Riera Quiles, en el que se informa sobre el



Nuevos productos Mitsubishi

Mabel, S. A., distribuidor oficial de Mitsubishi, ha presentado nuevos productos para MSX. Podemos destacar la aparición de dos nuevos modelos de ordenador; el ML-FX1 y ML-FX2. Ambos poseen 80 K de RAM y teclado numérico, además de incorporar 32 K de software en ROM (sólo la versión ML-FX2). Este paquete de aplicación incorporado en estas versiones, también se comercializará en cartucho para aquellos que posean la versión antigua o cualquiera que tenga un MSX. Los 5 programas de que se compone el MAP (Mitsubishi Applied Program —paquete de aplicación de Mitsubishi) son los siguientes; Base de Datos, Hoja Electrónica, Proceso de Textos, Gráficos y Comunicaciones. Sin embargo, hay que destacar la unidad de discos denominada ML-03FD con una capacidad de 720K formateados y 8 formatos diferentes de grabación. El punto fuerte





de esta unidad, es que viene preparada para añadir otra adicional. Efectivamente, en la carcasa, podemos apreciar una tapa que protege el hueco donde irá situada otra unidad de disco en caso de que el usuario lo desee. Teniendo en cuenta esta posibilidad y haciendo un cálculo rápido, podremos obtener un total de ¡1.4Mb de información!

Por último, cabe destacar la aparición de un ratón denominado ML-10 MA, muy útil para programas de gráficos y especialmente preparado para ordenadores de la 2.ª generación.

Para más información, dirigirse a:
MABEL, S. A.
Maragall, 120, Entlo. 1
08027 Barcelona

Tarjeta inteligente; ¿La solución final?

Frente a la piratería de software hay medios y medios, lo último se llama Tarjeta Inteligente o Bee Card, a la cual ya hemos hecho referencia en números anteriores. Esta vez, la comercialización de programas en este formato parece ser definitiva. Aunque actualmente existen dos formatos, el inglés (de la firma Electric Software) y el japonés (la conocida Bee Card), en nuestro país se comercializarán los dos tipos de formatos, de la mano de Serma, distribuidor en exclusiva de dicho producto. Pero la innovación, reside en el importante hecho que supone la fabricación de software a medida.

Efectivamente, la máquina en cuestión, permite pasar el master de un programa hasta un total de 10 tarjetas vírgenes con el programa que el usuario prefiera. Sin embargo, lo importante de esta medida es que dicho aparato graba toda la información necesaria para, una vez al mes y mediante una tarjeta de control, volcar vía RS-232C toda la información sobre el juego, el número de unidades vendidas, etc. Con esto, la piratería y las importaciones paralelas van a sufrir un serio revés al tener la compañía un control absoluto sobre todos los juegos que se comercialicen en tarjeta del mercado.

Idealogic también se apunta a MSX

Esta empresa, conocida por sus programas educativos y algunos buenos juegos, inicia su andadura en MSX. En colaboración con DIMension NEW, comercializa dos programas derivados de sendos juegos de mesa. Se trata del Dominó y las Damas, ambos en cassette y muy sencillos de utilizar. Por otro lado y desarrollado por la propia casa, posee un programa que facilita la labor de aprendizaje en la difícil tarea de escribir a máquina, denominado Idea type. En 12 lecciones, el usuario aprenderá y ejercitará la técnica mecanográfica.

Por otro lado, SONY comercializará los programas adaptados de esta empresa en cartuchos ROM. Estos programas, populares y bien acogidos por otros ordenadores personales, pronto verán la luz en el sistema MSX. Para ello, Idealogic ha adaptado los programas Compulandia, Teclas Divertidas, Mil Caras y Profesión Detective el estándar.

Además de estos programas, Canon será quien distribuya tres programas IDEA que esta firma pondrá en el mercado en breve. Los programas IDEA TEXT, DIM CALC y DIM BASE, vendrán en disco y convertirán el ordenador en una potente herramienta de trabajo.

Los periféricos también entran en los planes de esta empresa. Joystick II es una palanca de juegos sensible y sólida cuyo funcionamiento, al principio, resulta extraño principalmente debido a su recorrido.

movimiento, la composición interna y el recorrido del cometa Halley.

Al concurso se presentaron 210 programas: 164 de particulares (52 educativos, 55 de juegos y 57 de utilidades) y 46 de colegios. El jurado estuvo compuesto por Birguita Sandberg, Directora de la Editorial Manhattan Transfer; Pere Botella, Vicerrector de la Universidad Politécnica de Barcelona; Santiago Guillem, Director del Centre Divulgador de la Informática de la Generalitat de Catalunya; Fernando Landaluce, Gerente de Iveson Software, S. A. y Narcís Figueras, del departamento de Microinformática de Sony España.

Las bases del concurso otorgan a Sony España los derechos de compra sobre la totalidad de los programas presentados a concurso, derechos que Sony pudiera hacer valer en el caso de dos o tres programas que han despertado su interés.

El acto de la entrega de premios estaba previsto para el día 20 de enero, pero se vio retrasado a última hora por la ausencia de un alto cargo de la empresa que tenía que hacer personalmente la entrega.

La nueva tendencia del mercado, tiene en los ordenadores personales de 128 K su máximo exponente. La mayoría de los aparatos conocidos como Spectrum, Commodore y Amstrad, tienen una versión de 128 K en lo más alto de la gama.

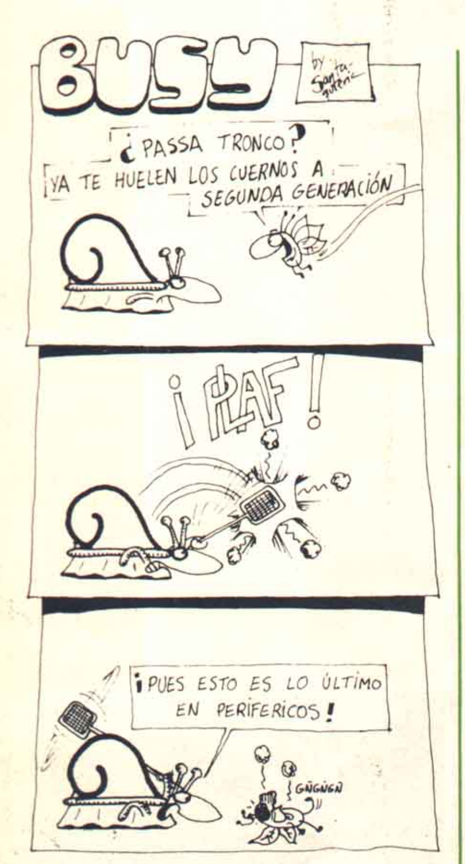
Con este paso, dichas casas pretenden romper el ritmo establecido por el mercado de los ordenadores personales, donde hasta el momento se habían impuesto los aparatos de 64 K (como el Commodore 64 o Amstrad) e incluso los de 48 K (como el Spectrum). Pero los fabricantes del estándar no van a perder el tren impuesto por el resto de las casas y en unos meses veremos hecha realidad la 2.ª generación de MSX. Estos se distinguen por unas mejoras difícilmente igualables por cualquiera de sus competidores, además de poseer 128 K de memoria.

En la actualidad, el marcado de ordenadores MSX se divide en dos apartados, donde el primero está cubierto por los ordenadores básicos, como el Goldstar FC-200, Sony HB-75P y HB-55P, Philips VG-8020 y VG-8010, Canon V-20, etc., ejemplos claros de la tendencia inicial. En el segundo apartado encontraremos unas versiones actualizadas de las primeras máquinas, como el Spectravídeo 738 X'press, Toshiba HX-22 y Pioneer PX-7 (estos dos se comercializa-





MSX II: El estándar contraataca



rán en breve). La diferencia básica entre estos grupos viene impuesta por unas mejoras que los segundos tienen con respecto a los modelos anteriores, como unidad de discos incorporada, tratamiento de textos, etc.

Los ordenadores mencionados están a mitad de camino entre la primera y segunda generación. En ellos no hay innovaciones importantes, el microprocesador, el chip de vídeo y el chip de sonido son los mismos, y salvo algunas excepciones, las máquinas permanecen inalteradas.

Sin embargo, lo que la 2.ª generación trae, es un cambio importante en el hardware, pero manteniendo la compatibilidad existente entre todos los equipos MSX, piedra angular de esta norma.

Es importante resaltar que el gran inconveniente de los equipos anteriores ha desaparecido en los nuevos aparatos, es decir, la resolución en el modo de textos ha variado y pasado a 80 columnas en lugar de las 40 iniciales, con lo que se mejora sustancialmente las posibilidades del ordenador, sobre todo, si se utiliza un tratamiento de textos, base de datos o cualquier programa de aplicación. Esto es importante, pero lo más práctico es, sin lugar a dudas, la posibilidad de que el ordenador pueda ejecutar programas en *CP/M*, cuya extensa biblioteca convertiría a nuestro aparato en una máquina semi-profesional, sin descuidar ni su origen ni a quien va destinado.



Foto 1: Philip YES, compatible PC con discos de 3,5 pulgadas, útiles en el MSX.

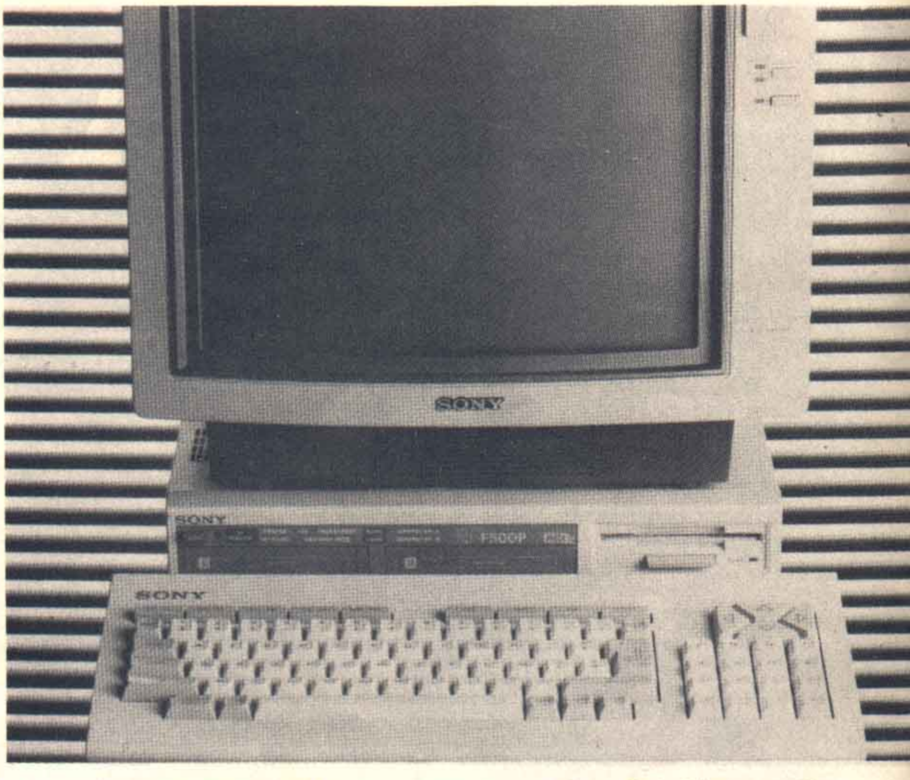


Foto 2: Sony, con su HB-F500P, es un claro representante de la 2.ª generación.

La resolución de la pantalla en caracteres a aumentado, al igual que el número de modos *SCREEN*. Estos han variado, pasando de 4 a 9 modos con distintas características que veremos más adelante.

El modo de alta resolución (*SCREEN* 2), se ha mejorado ostensiblemente, y donde antes se representaban 256x192 *pixels*, ahora se puede obtener 512x212. Obviamente, con estas cualidades, y con la posibilidad de utilizar hasta 16 colores (en este modo), se pueden crear auténticos juegos de fantasía, como esperamos ver algún día.

Sin embargo, en el modo de re-

solución más bajo, se dispone de una paleta de ¡256 colores!, lo que redundará en beneficio de los programas, tanto comerciales como caseros, de gestión o de juegos.

Esto sólo es el principio, puesto que la forma de crear y utilizar los *sprites* también se ha visto mejorada sensiblemente. Todo el que posea un MSX, sabrá que sólo se permiten representar en la pantalla hasta 4 *sprites* en una misma línea, lo que obligaba a realizar notables esfuerzos para intentar resolver el problema que esto representa, y aplicar la denominada regla del quinto *sprite*. Esta se ha modificado y se ha convertido en la regla del noveno *sprite*. Efectivamente, en la generación venidera los *sprites* y, en suma, la representación en pantalla, se ha mejorado notablemente. Se pueden definir y utilizar 8 *sprites* en la misma línea, lo que dará una mayor versatilidad, además de mejorar cualquier tipo de programa.



***Foto 3:** Kay Nishi, padre del estándar, apuesta por la generación venidera.*

Una de las mejores características que posee esta nueva generación, es la creación de *sprites* en diversos colores. Esta operación, en los ordenadores anteriores, se efectuaba utilizando varios *sprites*, es decir, si queríamos hacer un gráfico, bien sea una figura para un juego o cualquier elemento adicional de un programa, teníamos que definir dos *sprites* independientes con su color correspondiente, y luego, superponerlos en unas coordenadas concretas para que diese la sensación de que el gráfico es único. Con la 2.ª generación, se podrá definir el *sprite* por líneas, como se realiza en la actualidad, pero dando a cada una de éstas un color determinado, lo que no sólo redundaría en beneficio de la calidad de los

programas, sino que también permitiría mezclar colores, obteniendo unos resultados totalemente sorprendentes. Otro punto muy importante de estos equipos lo encontramos cuando tengamos que detectar colisiones y choques entre *sprites*. En la actualidad, cuando dos *sprites* chocan entre sí, el ordenador indica dicha colisión, mientras que en los MSX de la 2.ª generación, el ordenador no sólo indica que ha tenido lugar un choque, sino que nos dirá en que lugar de la pantalla ha ocurrido, lo cual significa que la labor de los programadores de juegos se ve enormemente simplificada al tener controlada totalmente la situación en un momento determinado.

No obstante, donde MSX juega una baza fundamental e importante es en la posibilidad de utilizar un magnetoscopio para leer, grabar y tratar cualquier tipo de información que se encuentre almacenada. Esto marca la diferencia entre los nuevos ordenadores y la supuesta competencia, ya que en este aspecto, la 2.ª generación de ordenadores MSX no da pie a ello.

Hoy por hoy, no existe ordenador personal en el mercado que ofrezca, además de una memoria de 128K y las características antes mencionadas, la posibilidad de controlar las señales generadas por una cámara de vídeo, una televisión o un magnetoscopio, con lo que se convierte en una diferencia muy importante a la hora de elegir un ordenador personal.

Dos son las empresas dedicadas a compaginar las aplicaciones del vídeo con el ordenador doméstico; Philips y Sony. Este campo, aún en vías de desarrollo, tiene unas aplicaciones que van desde bases de datos hasta las aulas informáticas. No vamos a profundizar en este tema puesto que en los números de diciembre y enero ya expusimos una visión completa de las posibles aplicaciones y de las características técnicas que ofrece un aparato de vídeo o un *Compact Disc* conectado a un ordenador.

Dejando un poco de lado el aspecto gráfico de la nueva generación, otro apartado que se ha visto retocado, ha sido el del sonido. Es obvio que el chip AY-3-9810 ya está anticuado y, aunque sea el más extendido, tiene sus días contados. Su sustituto es el chip de sonido mejorado que incorpora el ordenador Yamaha, cuyo funcionamiento aún está por ver. Lo que sí está claro es que a partir de ahora, los juegos para ordenadores MSX no van a ser los mismos.

Llegados a este punto, algunos lectores se preguntarán, no sin razón, si con tantos cambios realizados no hubiera sido mejor diseñar un ordenador personal con un microprocesador de 16 *bits*. Sin lugar a dudas, el chip Z-80 de Zilog, lleva muchos años en el mercado, pero parece ser que todavía le quedan unos cuantos más. El futuro de los MSX está en conseguir un ordenador con un chip de 16 *bits*, un Z-800 ó Z-8000, con lo que la efectividad del aparato se vería tremendamente aumentada. Ambos casos se están estudiando, y por el momento se ha descartado el Z-8000, puesto que plantea problemas de incompatibilidad con el chip Z-80, base de los ordenadores del estándar, algo que no debe ocurrir bajo ningún concepto. El Z-800 es el que más posibilidades tiene, pero, por el momento no existe en cantidades suficientes que permitan asegurar su futuro a medio y largo plazo. De cualquier manera, se rumorea que a finales de este año recién comenzado, podramos tener alguna noticia (los primeros MSX de 16 *bits* en Japón) de los avances llevados a cabo.



Foto 4: Spectravideo X'press, a medio camino entre las dos tendencias.

No sólo se está trabajando en

este chip para lograr dar el salto a los microprocesadores de 16 *bits*. También se está teniendo en cuenta la posibilidad de utilizar dos microprocesadores y utilizar cada uno según el caso. Bajo este aspecto, todos los indicios se vuelcan hacia el chip 68000 de Motorola, el cual está teniendo un auge sin precedentes en el mercado de los ordenadores personales, ya que es el corazón de máquinas tan conocidas como el Apple Macintosh y el Atari 520ST. Lo más seguro es que en las futuras versiones de MSX, o digamos, la tercera generación, nos encontremos con dos microprocesadores en una máquina, el 68000 de Motorola para potenciar el ordenador más si cabe, y el Z-80 de Zilog para mantener la siempre importante compatibilidad.



Foto 5: La cámara de vídeo se convertirá en un periférico más.

Las especificaciones técnicas de la 2.ª generación de MSX no han cambiado, siendo sus características esenciales las siguientes:

– El chip *UCP* sigue siendo el Z-80 de Zilog, funcionando a una velocidad de 3,6 Mhz.

– La memoria *RAM* contará con 64K (mínimo), mientras que la *VRAM* tendrá 128K.

– El chip de vídeo permitirá mostrar hasta 9 modos de pantallas, que son los siguientes:

Modo texto de 40 car. x 24 líneas.

Modo texto de 80 car. x 24 líneas.

Modo texto de 32 car. x 24 líneas.

Modo gráfico de 256 x 192 pixels, con 16 colores.

Modo gráfico de 64 x 48 pixels, con 16 colores.

Modo alta resolución de 256 x 212 pixels, con 16 de 512 colores.

Modo alta resolución de 512 x 212 pixels, con 4 de 512 colores.

Modo alta resolución de 512 x 212 pixels, con 16 de 512 colores.

Modo alta resolución de 265 x 212 pixels, con 256 colores.

Como podemos ver, se dispone de una paleta de 512 colores aunque en el mejor de los casos podremos utilizar 256, y hasta 8 *sprites* pueden aparecer en una línea.

– Todos los ordenadores incorporan una unidad de superposición de vídeo y una unidad de discos de 3.5 pulgadas, lo que significa que el *BASIC* también se verá afectado por esta nueva mejora. El lenguaje *BASIC* se verá ampliado por la aparición de las instrucciones necesarias que controlarán vídeos, aparatos de *Compact Disc* o imágenes de televisión. Algunas de las instrucciones que encontraremos son:

– COLOR = (NUM. DE PALETA, ROJO, VERDE, AZUL), que cambiará la tabla del color y especificará la intensidad de los tres colores primarios.

– COLOR SPRITE (NUM. PLANO), (EXPRESION), esto especifica el color de cada línea que compone un *sprite*.

– COLOR SPRITE (NUM. PLANO) = (NUM. DE PALETA), especifica un color para un *sprite*.

– SET VIDEO, prepara el modo de superposición.

– COPY VIDEO, digitaliza una imagen exterior de vídeo.

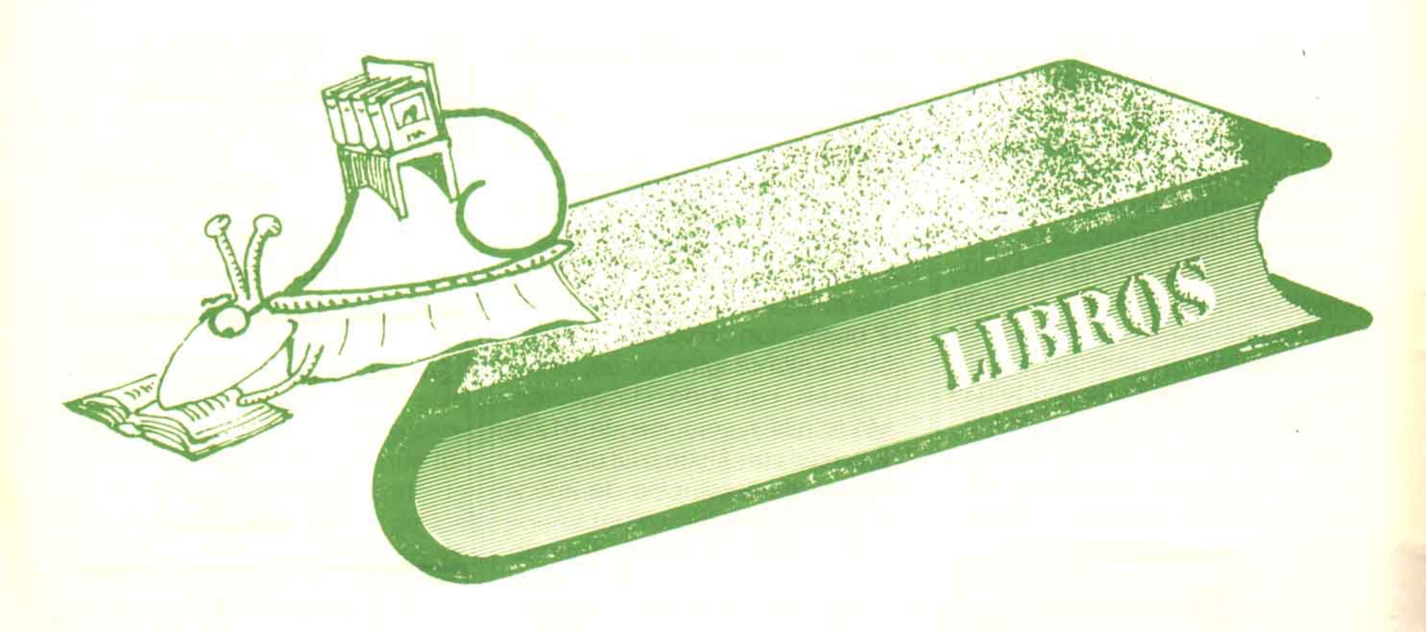
– GET DATE, GET TIME, lee la fecha y la hora del reloj del sistema.



– SET DATE, GET TIME, modifica la hora y la fecha del reloj del sistema.

No cabe la menor duda que estas mejoras van encaminadas a cubrir un pequeño espacio dentro del gran mercado de ordenadores personales y más dentro de la gama MSX; y su precio, sin llegar a ser prohibitivo, va a jugar una baza fundamental. Podemos esperar que con los nuevos ordenadores, se inicie una etapa hasta ahora desconocida, en el ámbito del ordenador personal.

Hasta la fecha, dos grandes fabricantes, como Philips y Sony has dado el paso de gigante hacia la 2.ª generación. El primero con un ordenador denominado VG-8230, y el segundo con el HB-550P, ambos con buena presencia, con algunas características expuestas a lo largo de estas páginas y esperando la ocasión ideal para iniciar su andadura.



Título: Inteligencia artificial: conceptos y programas
Autor: Tim Hartnell
Editorial: Anaya Multimedia
Páginas: 267

El mundo de los ordenadores nos lleva, por momentos, a una aventura fascinante en un reino donde se difuminan las barreras entre el hecho científico y la ciencia-ficción: la inteligencia artificial es un hecho hoy en día.

¿Puede la mâquina pensar realmente?, ¿la inteligencia es natural o no? Vamos a descubrir en este libro todos los avances científicos y la investigación realizada en torno a todo este sin fin de preguntas y teorías.

Este libro es un manual de experiencias frente al ordenador, en el que las muestras de inteligencia aparecen respondiendo a situaciones, tomando decisiones y actuando de acuerdo con éstas. No es que el ordenador sea «consciente de sus actos», sino que podemos ser capaces de crearle al ordenador inteligencia (artificial).



Los apartados de este libro se refieren a programas que razonan y aprenden, pero el fin primordial se encuentra en la investigación.

Todos los ordenadores tienen capacidad para crear, utilizando como medio nuestra inteligencia, por ello, en el apartado «Arboles», se utiliza el método inductivo para llevar al ordenador a una conclusión.

Otro apartado que ha tratado extensa y objetivamente el libro que nos ocupa, es el referido a la comprensión del lenguaje natural. Bloquelandia es la traducción de un programa que nos permitirá experimentar la capacidad del ordenador para comunicarse en nuestro lenguaje, y así otorgarle un grado mínimo de inteligencia. El campo de la lingüística informática, tradicionalmente se ha dedicado a la investigación de los modos de análisis de la oración, en esperanza de que la máquina, ante unas bases prefijadas, al analizar la oración, pueda comprenderla.

Cuando pensamos en las posibilidades que surgen del hecho de que las máquinas sean capaces de comprender y procesar en una lengua natural, no parece una extravagancia el imaginar que estas mismas podrían resultarnos de gran ayuda a la hora de traducir de una lengua a otra.

No sólo este libro se ha preocupado de introducirnos en esta fantástica búsqueda, sino que su carácter didáctico nos ayuda a mejorar las técnicas de programación: corrección de programas, planificación, documentación y otras muchas actividades reflejadas en sus apéndices.

Un libro experimental no sólo basa sus teorías en sus propias experiencias, sino que se ayuda de otros manuales que puedan aumentar su contenido, por lo que su autor le ha complementado con una bibliografía referida a este tema, y ha creado un glosario de términos informáticos. Por esto, no sólo es un libro innovador, sino que es el canalizador de una serie de teorías y prácticas para asentar las bases de la informática en el apartado referido a la inteligencia artificial.

Título: Programación Básica. MSX.
Autor: Jonathan Pearce y Graham Bland
Editorial: Paraninfo
Páginas: 171

En el mundo del microordenador doméstico hacía falta una compatibilidad demostrada, que hiciese real las ilusiones de muchos usuarios de poder intercambiar elementos de diversas características.

Llegó MSX, y con él finalizaron los problemas de incompatibilidad de hardware y periféricos, y lo más importante: un Basic común para todos.

Podemos decir que la entrada en el mercado de MSX, ha supuesto un hito en la historia, la estandarización está llegando a todos los ámbitos y la industria japonesa está imponiendo sus bases, incluso en mercados que le estaban vetados: Reino Unido, E.E.U.U., etc.

Por todo ello, aunque existan numerosos libros que sienten las bases de la programación en Basic para MSX, era necesaria la presencia de uno de ellos que hiciese referencia a todos los aparatos de este estándar y a sus creadores, por lo que pensamos que «Programación Básica. MSX.», puede ser el libro adecuado para los usuarios del sistema MSX.

En el libro se hace una presentación detallada de todos los ordenadores que se rigen dentro de estas normas, sus elementos, ampliaciones y su estructura interna, para que así podamos saber todas las posibilidades con las que cuenta nuestro aparato, y aquellas que en un futuro próximo van a llegar.

Apartados dedicados al trabajo con números (constantes, variables, etc.), junto con las expresiones numéricas, nos harán conocer la multitud de funciones que podemos realizar y la forma más fácil de conseguirlo.

Música y sonidos, quizá nos parezca que este apartado no tenga importancia, pero es que no está concebido en el aspecto tan sencillo en que nosotros lo clasificamos. Un generador de sonidos sirve para emitir sonidos, incluso puede, y de hecho ya se ha conseguido, imitar la voz humana, pero para lograr todo esto hay que conocer el lenguaje de su aparato, y éste es el Basic, un Basic abierto a todos y comprensible por todos, por lo que resulta necesario que nos sirvamos de un libro que nos ayude a conseguir todo esto y además nos enseñe.



No es un libro práctico en todo el contenido de la palabra, ya que entonces tendrían que haber editado otro volumen, pero han sabido decir las cosas y decirlas bien, y eso es lo importante.

Todos los principios del lenguaje Basic se encuentran señalados y estudiados de una forma amena y didáctica, sin duda es todo un tratado de informática para el sistema MSX. Todo esto hace que sea una obra muy útil, tanto para principiantes, como para programadores que quieran profundizar en el revolucionario MSX.

Tratamiento de datos (II)



Gracias al ordenador, los cálculos, por complicados que sean, se simplifican de una manera importante siempre y cuando tratemos los datos adecuadamente. El lógico que cualquier tipo de error por nuestra parte, lleve a la obtención de resultados totalmente equívocos, debido principalmente a un mal planateamiento inicial.

Supongamos que somos un físico experimental de mediados del siglo pasado y que tras algunas conjeturas y no menos ratos de paciente reflexión estamos convencidos de que la resistencia eléctrica de un conductor metacilíndrico viene dado por la fórmula:

$$R = \varrho \frac{l}{S} \qquad (a)$$



R: Es la resistencia en ohmios.

l: La longitud del conductor en metros.

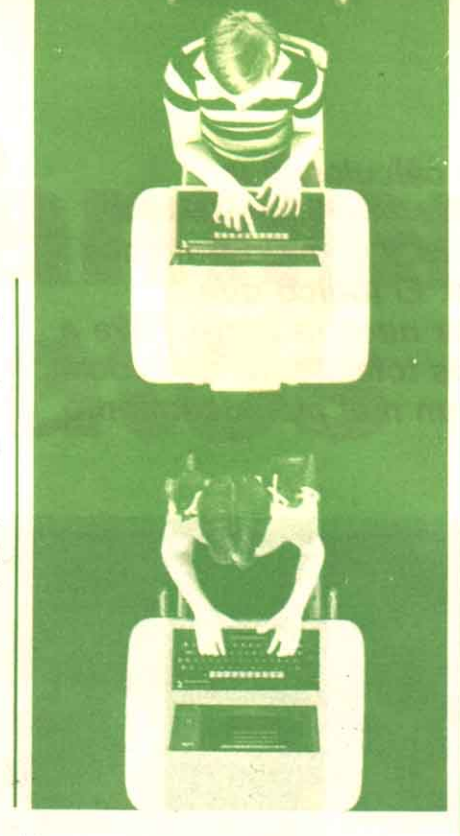
S: El área del conductor en milímetros cuadrados.

ϱ: Un coeficiente que varía con la temperatura.

Asimismo pensamos que la variación del coeficiente con la temperatura viene dado por la fórmula:

$$\varrho_T = \varrho_{20}\ [1 + \alpha_T(T - 20)] \quad \text{(b)}$$

donde:



ϱ_T: es el coeficiente que aparece en la fórmula del párrafo anterior a la temperatura T (expresada en grados centígrados).

ϱ_T: Es el valor de ϱ medido a 20°C.

α_T: Es un coeficiente que depende sólo del metal con el que está construido el conductor.

T: Es la tamperatura medida en grados centígrados.

Las fórmulas indicadas anteriormente no son más que una forma de expresar matemáticamente de forma concisa y abreviada las hipótesis que tras utilizar creativamente nuestra imaginación hemos establecido, no son más que suposiciones o conjeturas, y como somos un físico serio y riguroso, nos disponemos a consultar el único juez que «entiende» las leyes de la naturaleza: la experien-

INTENSIDAD MEDIDA EN LAS MUESTRAS (Amoerios)

Muestras (1 a 9)

Temp.

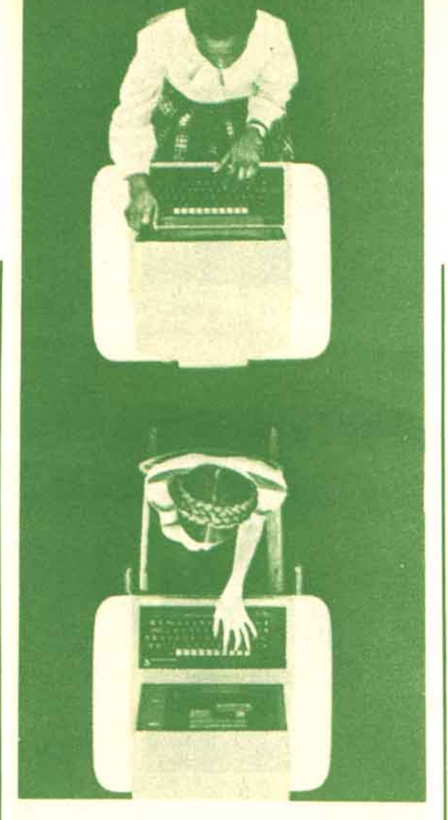
ºC \ M	1	2	3	4	5	6	7	8	9
10	0.73	1,28	2,52	1,66	3,15	0,45	4,31	0,53	1,08
15	0,77	1,34	2,69	1,67	3,22	0,46	4,11	0,56	1,05
20	0,77	1,28	2,69	1,75	3,07	0,44	4,33	0,54	0.98
25	0,73	1,16	2,48	1,66	3,22	0,44	4,31	0,50	1,02
30	0,70	1,19	2,46	1,61	3,05	0,41	4,32	0,51	0,90.
35	0,68	1,19	2,25	1,62	3,03	0,40	4,20	0,49	0,93
40	0,72	1,21	2,25	1,57	2,83	0,43	3,91	0,52	0,96
45	0,68	1,17	2,43	1,45	2,97	0,37	3,69	0,50	0,86
50	0,67	1,04	2,30	1,48	3,03	0,39	3,94	0,50	0,86
55	0,62	1,10	2,32	1,37	2,79	0,36	3,67	0,51	0,85
60	0,63	1,04	2,11	1,48	2,67	0,40	3,91	0,45	0,84
65	0,60	1,01	2,28	1,50	2,74	0,37	3,59	0,47	0,83
70	0,59	1,00	2,15	1,35	2,58	0,38	3,78	0,49	0,84
75	0,59	1,09	2,00	1,45	2,79	0,35	3,63	0,45	0,85
80	0,63	0,95	1,97	1,39	2,54	0,33	3,44	0,42	0,76
85	0,58	0,96	1,95	1,36	2,64	0,34	3,33	0,46	0,81
90	0,57	1,03	2,05	1,38	2,52	0,36	3,56	0,44	0,78
95	0,58	0,98	1,94	1,36	2,39	0,35	3,47	0,41	0,79
100	0,56	0,94	1,83	1,28	2,32	0,32	3,36	0,44	0,76

Tabla 1



cia, el experimento físico. Si comprobamos que tendremos que efectuar, habremos arrancado otra ley a la naturaleza. En el caso de que las medidas obtenidas no estén de acuerdo a los resultados predichos por las fórmulas que constituyen la base de nuestra teoría, deberemos seguir buscando para establecer otras hipótesis que sean avaladas por la experiencia.

Pues bien, llegado este punto no nos queda otra solución que llevar a cabo un experimento con las adecuadas garantías, en lo que a la validez de las medidas se refiere, para comprobar si nuestras suposiciones son verdaderas o no. Para ello, de alguna manera, conseguimos 3 trozos de hilo de cobre de 60 metros de longitud y



de diámetros respectivos 0,5; 0,75 y 1,2 mm., cortamos cada uno de los trozos en longitudes de 15,20 y 25 metros con lo que disponemos de 9 muestras tal como aparecen en la tabla 1.

Disponiendo ya de las 9 muestras de hilo y tras pedir prestados algunos aparatos de medida montamos un banco de medidas tal como se indica en la figura 2. Esencialmente no es más que un circuito de corriente continua donde medimos la tensión y la intensidad que atraviesa los hilos que han sido sumergidos en un baño de aceite para controlar su temperatura. Nuestra manera de proceder será la siguiente:

1) Ajustaremos la llama del quemador hasta leer en el termómetro la temperatura en la que estemos interesados.

2) Conectaremos el amperímetro a la muestra número 1 y ajustaremos el mando del reostato

hasta que el voltímetro marque 1 voltio. En este momento si llamamos I a la indicación del amperímetro, la resistencia del conductor será 1/I. Anotaremos el valor de la muestra y la temperatura de la medida.

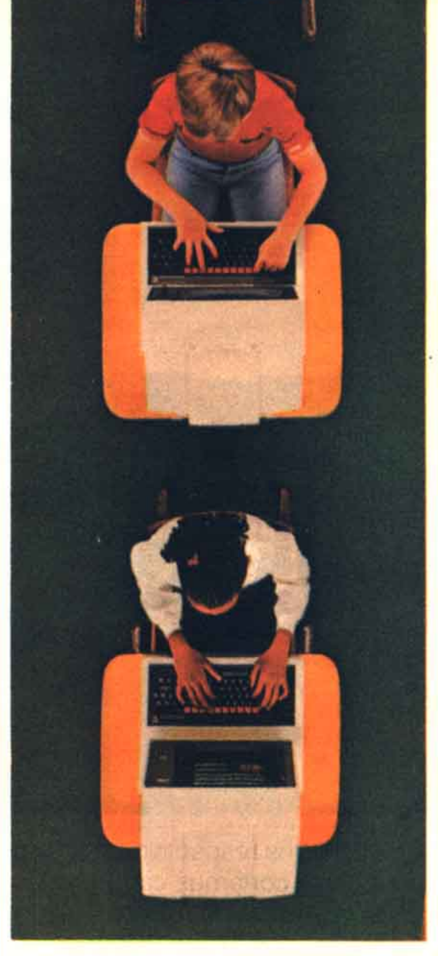
3) Repetimos el proceso del punto anterior para las 9 muestras.

4) Efectuamos medidas tal como se indican en los puntos (1) y (2), para una serie de temperaturas diferentes.

En la tabla 2 se muestran unos supuestos resultados de medidas efectuadas entre 10°C y 100°C.

Nos queda finalmente la verdadera verificación del cumplimiento de las relaciones (a) y (b) que son la base de nuestra teoría. Para ello vamos a utilizar el siguiente procedimiento:

1) Para cada muestra y temperatura de medida podemos calcular ϱ utilizando para ello la fórmula (a).



2) Puesto que le ecuación (b) no es más que una relación lineal entre ϱ y T, relación que matemáticamente puede expresarse como la ecuación de una recta, es de esperar que si nuestras hipótesis son válidas y por tanto las expresiones (a) y (b) correctas, las parejas de valores ϱ_T/T obtenidas en las diversas muestras y a diferentes temperaturas durante el experimento deben de estar situadas sobre dicha recta.

3) Para medir matemáticamente hasta qué punto se aproxima un conjunto de puntos a una recta, existe un parámetro denominado coeficiente de correlación. Se dispone de un procedimiento muy detallado y sistemático para calcular dicho coeficiente de correlación de las parejas ϱ_T/T.

4) La situación es tal que cuanto más se aproxime el coeficiente

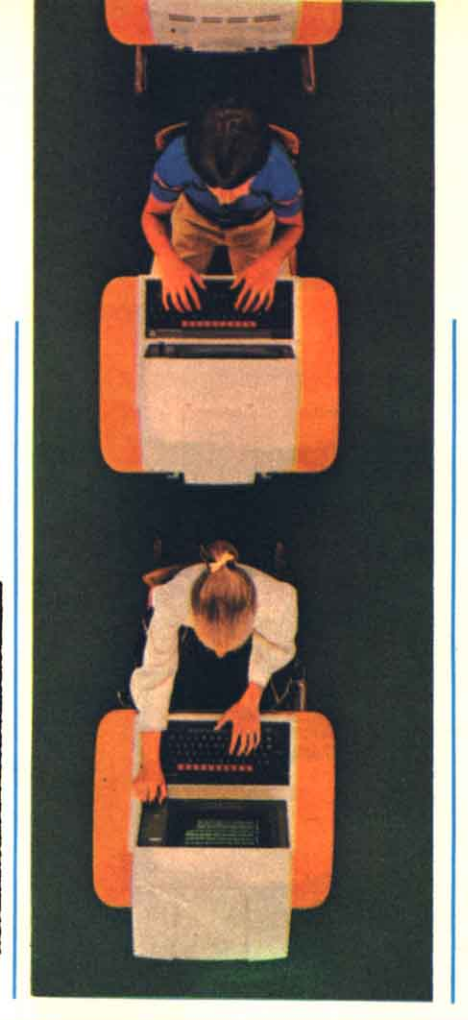
A
REOSTATO
V
TERMOMETRO
HILOS DE COBRE
CONTENEDOR CON ACEITE
QUEMADOR

Figura 1

de correlación a 1 mayor «proximidad» existe entre el conjunto de puntos y la recta. Por tanto ya disponemos de la clave para verificar la validez de nuestra teoría: El coeficiente de correlación calculado debe de ser próximo a 1.

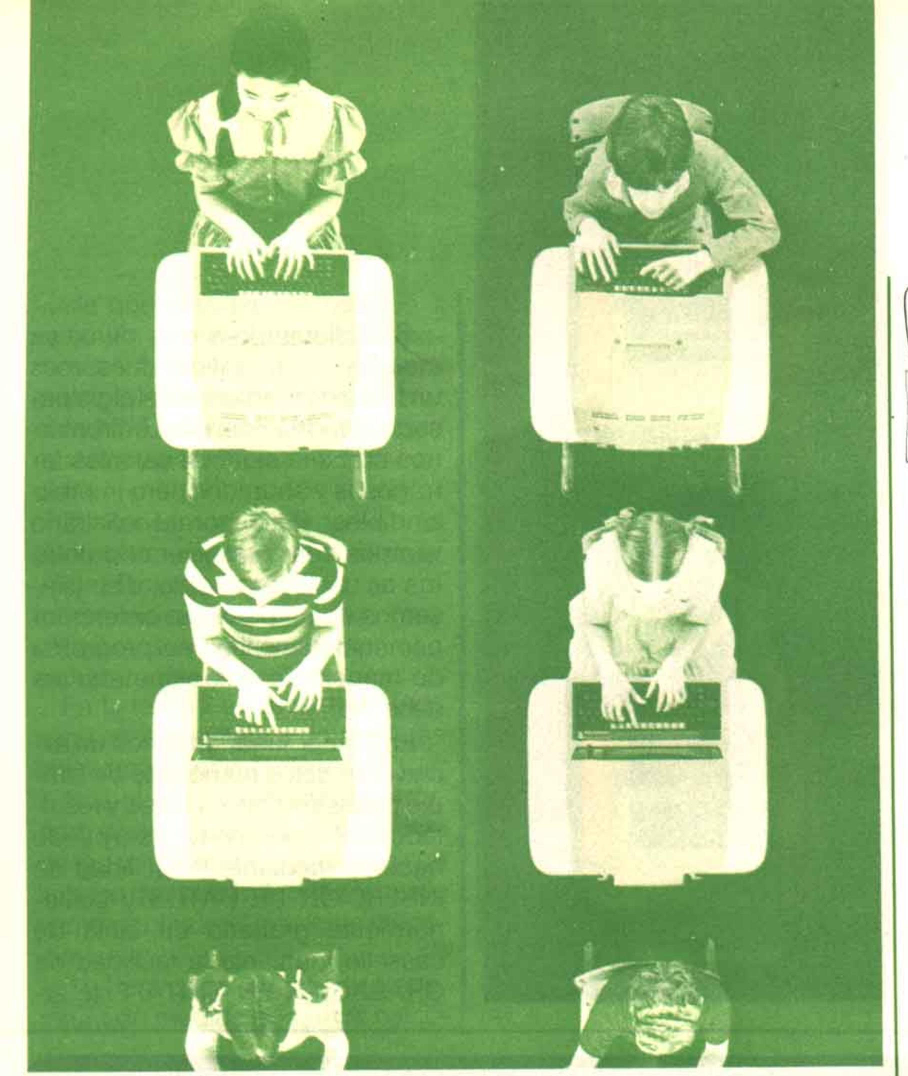
muestra nº	longitud (mts)	diámetro (mm)
1	15	0,50
2	20	0,75
3	25	1,20
4	15	0,75
5	20	1,20
6	25	0,50
7	15	1,20
8	20	0,50
9	25	0,75

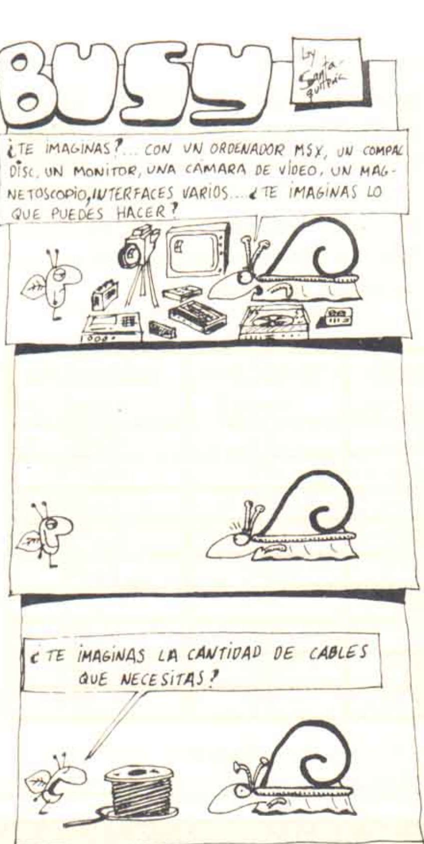
Tabla 2



Bien, llegando a este punto es cuando, si en realidad fuésemos un físico de mediados del siglo pasado, tendríamos que enfrentarnos con una serie de cálculos farragosos y aburridos pero imprescindibles. Pero por el contrario veamos como resolver el problema si, misteriosamente, dispusiésemos de un pequeño ordenador personal doméstico y el programa de manejo de datos que aquí se describe:

En primer lugar creamos un archivo de datos numéricos de simple precisión con los datos y resultados del experimento. Esto puede hacerse mediante la facilidad de INSERCION DE DATOS y posteriormente grabarlo en cinta de cassette mediante la facilidad de GRABACION EN CINTA. Este ar-



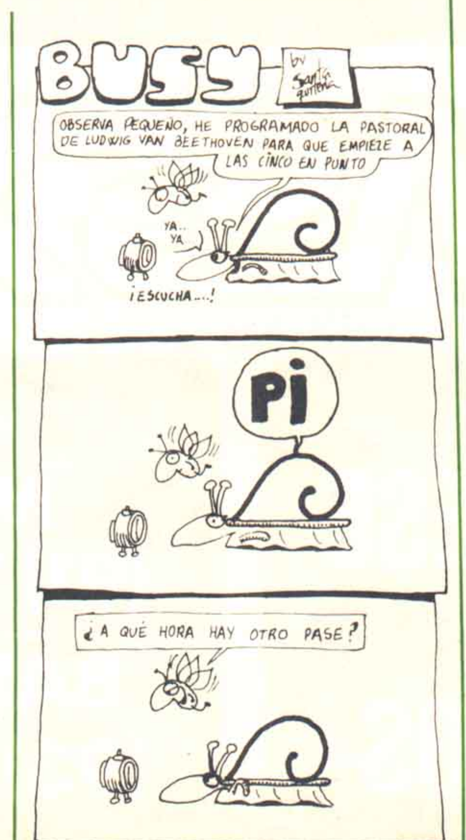


1.° 1 registro de 18 datos (filas 1 al 18) donde aparezcan ordenados la longitud y el diámetro de las 9 muestras (15; 0,50; 20; 0,75; 25; 1,20;...).

2.° Sucesivos registros de 10 datos, 1 por cada temperatura de medida, y en los que aparezcan como primer dato la temperatura de la medida y los 9 datos en la muestra 1 a 9. Estos registros se situarían a partir de la línea 19 (en el caso de introducir los datos de la tabla 2, puede comprobarse que los datos correspondientes a la temperatura de 30°C ocupan las filas 59 a 68 en el siguiente orden 30; 0,70; 1,19; 2,46; 1,61; 3,05; 0,41; 4,32; 0,51; 0,90).

3.° Escribiremos un programa chivo puede tener la siguiente estructura:

que sea capaz de leer los datos del archivo que se acaba de crear, calcule el coeficiente de correlación y adicionalmente los valores estimados de ϱ_{20} y α_1 (un ejemplo de programa para este fin, que puede utilizarse junto con los datos de las tablas 1 y 2 para verificar el ejemplo que se expone y ejercitarse en la utilización del programa de manejo de datos, se lista tras el programa principal).



Finalmente, desde el punto de vista del físico decimonónico que estamos encarnando, solamente nos queda comprobar el valor del coeficiente de correlación para verificar o no la validez de nuestra teoría, no obstante, y dado que disponemos de tan asombrosas facilidades de cálculo, podemos permitirnos el lujo de efectuar nuevas medidas, que añadiremos fácilmente a nuestro archivo, para aumentar la fiabilidad del experimento.

Jorge Moreno Camacho

```
10 'CAL. RESISTIVIDAD/COEF.TEMPERATURA
20 '
30 '28/SEP/85  Ver.01
40 CLS
50 OPEN "CAS:RESULS" FOR INPUT  AS#1
60 FOR I=0 TO 8
70 INPUT#1,L(I),D(I)
80 PRINT L(I);D(I)
90 NEXT
100 CLOSE#1
110 PRINT:PRINT:PRINT
120 INPUT"NUMERO DE TEMPERATURAS DIFERENTES EN   LAS QUE SE HA EFECTUADO LA MEDI
DA :";NM
130 CLS
140 FOR I=1 TO NM
150 OPEN"RESULS" FOR INPUT AS#1
160 INPUT#1,T:PRINT T
170 FOR K=0 TO 8
180 INPUT#1,IN(K):PRINTIN(K)
190 NEXT
200 PRINT
210 ST=ST+9*T
220 T2=T2+9*T^2
230 FOR J=0 TO 8
240 RE=3.141592654#*D(J)^2/400000!/L(J)/IN(J)
250 SR=SR+RE
260 R2=R2+RE^2
270 TR=TR+T*RE
280 CLOSE#1
290 NEXT J,I
300 A=(TR-ST*SR/9/NM)/(T2-ST^2/9/NM)
310 B=(SR-A*ST)/9/NM
320 CO=A*SQR((T2-ST^2/9/NM)/(9*NM-1))/SQR((R2-SR^2/9/NM)/(9*NM-1))
330 CLS
340 PRINT"RESULTADO DE LA ESTIMACION":PRINT:PRINT
350 PRINT"Resistividad a 20Cº:";20*A+B:PRINT
360 PRINT"Coeficiente de temperatura de la re-   sistividad:";A/(20*A+B):PRINT
370 PRINT"Coeficiente de correlación resultante  de efectuar la regresión lineal
:";CO
380 END
390 CO=A*SQR((T2-ST^2/9/NM)/(9*NM-1))/SQR((R2-SR^2/9/NM)/(9*NM-1))
```

Programa: Soccer
Tipo: Juego
Distribuidor: Konami
Formato: Cartucho

¿Por qué esperar a que lleguen los mundiales?, Konami, con este juego, ha llegado en el momento oportuno. Ya no es necesario esperar a los encuentros del Mundial para ver a nuestra selección ganar todos los partidos, es más, está en nuestras manos ser los campeones o no.

En este juego para ordenador podemos comprobar la espectacularidad del mundo futbolístico, las tácticas que debemos emplear para vencer a nuestros contrarios, y la satisfacción que da la victoria, ante un público entusiasta que nos anima constantemente.

Las reglas de este juego son de todos conocidas, pero hay algunos aspectos referidos a la forma de jugar en este mundial que nos serán de gran ayuda.

Podremos elegir el color de nuestras camisetas y del equipo contrario. Así, en momentos de lucha por el esférico, sabremos a quién pasar directamente el balón sin temor a equivocarnos. Nuestros jugadores han sido creados a imagen de los grandes campeones, por lo que nos resultará muy agradable presenciar el encuentro, tanto de jugadores como de espectadores.

El color, se encuentra en su máxima expresión, ya que teniendo como fondo el intenso verde del campo futbolístico, resalta aún más los colores que defendemos y los de nuestros adversarios.

La realización gráfica del juego, se ve a la hora de representar lo más fielmente el cuerpo humano de los deportistas, que son el centro de nuestra atención. También se hacen notar estos gráficos en las instalaciones deportivas y en la



cancha donde se desarrolla el juego.

¿Quién será el árbitro de nuestro encuentro?, la máquina será quien decida, pero no a su favor. La arbitrariedad se verá patente a la hora de aplicar el tan temido fuera de juego, ya que sólo jugando en el nivel tres o superiores, se tendrá en cuenta. Las faltas directas e indirectas se pitarán en el momento de cometerse y ¡no existe ley de la ventaja!

En caso de que hubiese empate, se decidirá por penaltys, por lo que el control de nuestro portero sería un dato importante, pues en ese momento es cuando nuestra habilidad debe ser mayor.

Pero este juego no es un simple juego de fútbol, ya que como en la realidad, los contrarios pueden «sufrir» si nuestra forma de arrebatarles la pelota no es muy técnica.

El encuentro se divide en dos partes, pudiendo elegir entre 5, 10 ó 15 minutos cada parte.

Ahora tan sólo se trata de conseguir nuestro título, y lo principal, jugar.

Puntuación:
Presentación: 8
Claridad: 8
Rapidez: 9
Adicción: 9

Programa: Boxeo
Tipo: Juego
Distribuidor: Serma
Formato: Cartucho

Boxeo será para muchos el fiel reflejo de sus ilusiones o su mayor afición elevada al ordenador. Su calidad como juego, sus posibilidades, sus sprites (dibujos), y todos los pormenores del inigualable mundo del ring se encuentran aquí.

No es un juego que nos hará vivir en nuestra propia «carne» sus señales, pero psicológicamente nos hará vivir más de una sensación y efectuar movimientos de defensa y lucha mientras jugamos. Nuestro equipo de redacción «afirma» esta postura.



Como la técnica del *Boxeo* requiere, es necesaria una absoluta coordinación de movimientos y una concentración elevada, lo que unido a una alta preparación física nos permitirá luchar y pelear con los campeones mundiales de Konami. No, no es necesario que controle su dieta antes de jugar, pero un pequeño espacio de concentración no le vendrá nada mal.

Sus oponentes, seleccionados de todos los países del mundo, llegaron a la meta con las más temibles técnicas y los más efectivos golpes para lograr que usted no se encuentre entre ellos, pero no nos vamos a dejar batir tan rápidamente.

Fighting Fyu serás tú, y alcanzarás todas las cimas del mundo del *Boxeo* venciendo a todos los oponentes que han pisado jamás los cuadriláteros.

Por tanto, debemos conocer a nuestros contrincantes de una manera exhaustiva: *Red Wolf,* el mejor boxeador que tiene Inglaterra, sus rápidos movimientos harán que más de una vez le perdamos de vista; *More Betteryo Alí,* el gigante americano, ¡peligro!; *Sanchess,* el millonario italiano, que no sólo emplea golpes para ganar; *China Kan,* el serio y combativo boxeador chino convertido recientemente al lamaísmo; y por último, *Moai Junior,* el genial y gigantesco boxeador que siempre sorprende a sus múltiples seguidores, y a muchos de sus contrincantes de primera fila. Usted quedará atrapado desde el primer momento con las rápidas acciones y sorprendentes ataques de este juego.

Pero nuestra técnica es inigualable, rodeada de todo el ambiente de éxito que nos rodea, pasaremos un buen rato y desarrollaremos todo un torneo casero, obteniendo el título de «campeones de *Boxeo*».

Puntuación:
Presentación: 9
Claridad: 8
Rapidez: 9
Adicción: 9

Programa: Laberinto Espacial
Tipo: Juego
Distribuidor: E. M. S. A.
Formato: Cartucho

Entre el sin fin de juegos que podemos encontrar en el mercado, existen algunos específicos basados en la ficción, otros que nos sumergen en un mundo de fantasía y otros que, además de enseñarnos, nos hacen pasar un rato inigualable.

Laberinto Espacial nos hará pasar momentos de incertidumbre y miedo, y también nos hará esforzarnos para conseguir la clave que resuelva el enigma, es decir, poder salir del Laberinto.

Tampoco será el tipo de enigma que nos traerá de cabeza durante todo el juego, ya que más que resolverlo, tendremos que ser hábiles y rápidos para destruir las naves que nos acosan y encontrar el punto de nuestro destino: hallar el tesoro. Un tesoro secreto FUERZA, que controla toda la energía cósmica, el cual está escondido en las profundidades del *Laberinto Espacial,* protegido por naves destructoras y sin piedad. Tiene que conseguir subir a bordo de su nave y conducirla hasta el tesoro, atravesando todos los peligros que se encuentren a su paso y llegar hasta FUERZA, ahí encontrará las claves para conseguir el tesoro.

Esto será tarea ardua y difícil, ya que verse inmerso en un Laberinto le hará sentirse atrapado durante los dieciséis planos de que consta el juego, y además tendrá que anular o batir las numerosas naves que le perseguirán, ya que sino le derribarán.

Los elementos a tener en cuenta para desarrollar correctamente el juego se encuentran en usted, pero dependiendo de si utiliza joystick o los cursores de su aparato, deberá elevar la precisión de sus movimientos.

Cada vez que consiga llegar a FUERZA tendrá acceso al siguiente Laberinto de la nebulosa por la que se encuentra volando. Sus enemigos escondidos en todos los lugares saldrán a su paso cuando menos se lo espere, des-



de las estrellas, atravesando las paredes del Laberinto, etc.

Contará, no obstante, con una barrera roja que le servirá para destruir todas las naves que pasen por su lado con «malas» inten-



ciones, pero esto hará que consuma más energía de la que usted tiene, por lo que cada vez que encuentre una estrella roja deberá dispararle un misil, y así elevará su capacidad de energía, cuando necesite de nuevo energía un pitido leve sonará anunciándoselo.

El juego consta de tres naves y muchos peligros, pero cualquier dura tarea podrá ser superada fácilmente, sólo hace falta valor y... ¡suerte!

Puntuación:
Presentación: 7
Claridad: 7
Rapidez: 6
Adicción: 8

Programa: Boulder Dash
Tipo: Juego
Distribuidor: ERBE
Formato: Cassette

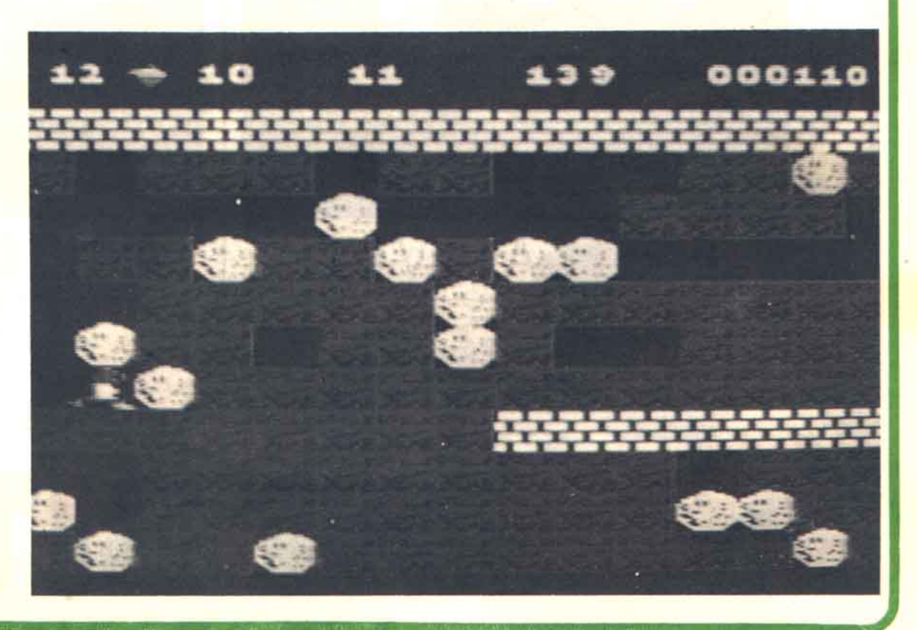
¿Qué te parece convertirte por unos momentos en un verdadero buscador de diamantes y correr todas las aventuras que ello conlleva?

Boulder Dahs es un juego en el que necesitarás de toda tu pericia y valor para guiar a *Rockford,* nuestro héroe, a través de 84 laberintos diferentes. Pero el problema no radica únicamente en encontrar la salida, sino que te encontrarás a tu paso con un montón de peligrosos enemigos que te impedirán desarrollar tu tarea. Moscas mortales, amoebas y un sin fin de elementos harán que pases apuros, pero esto no te debe asustar, ya que en el momento más insospechado te encontrarás con una joya inigualable que tendrás que coger, y asi conseguirás finalizar tu tarea.

Esta aventura te resultará muy divertida por los elementos con los que vas a contar; podrás excavar más rápido y puedes también utilizarlas como soporte para saltar a otro lugar, pero con cuidado: si quedas atrapado entre las rocas, morirás.

El juego se desarrolla muy rápido, y deberás planificar y dirigir exactamente tus movimientos, sino no conseguirás llegar a la meta.

Te moverás por cuatro cuevas en las que hay un número definido de joyas, las debes coger y así pasarás a la siguiente prueba. Será



Programas Sony MSX, para lo que guste ordenar.



JUEGOS



GESTION



EDUCATIVOS



APLICACIONES





PROGRAMAS SONY MSX

Educativos

- Monkey Academy
- Alfamat
- Viaje Espacial
- Multipuzzle
- Noria de Números
- Corro de Formas
- Coconuts
- Yo Calculo
- Selva de Letras
- El Cubo Informático
- Electro-graf
- El Rancho
- Teclas Divertidas
- Boing Boning
- Compulandia
- Mil Caras
- Logo
- Países Mundo-1
- Países Mundo-2
- Tutor
- Computador Adivino
- Aprend. Inglés-1
- Aprend. Inglés-2
- Cosmos
- Curso de Básic
- Juego de Números

Juegos

- Antártic Adventure
- Athletic Land
- Sparkie
- Juno First
- Car Jamboree
- Battle Cross
- Crazy Train
- Mouser
- Computer Billiards
- Alí Babá
- Track & Field-I
- Track & Field-II
- Dorodon
- Chess (Ajedrez)
- Senjo
- E.I.
- Lode Runner
- Super Tennis
- Backgammon
- Super Golf
- Hustler
- Binary Land
- Driller Tanks
- Stop the Express
- Ninja
- Les Flics
- La Pulga
- The Snowman
- Cubit
- Pack 16K
- Fútbol
- Kung Fu
- Batalla Tanques
- Mr. Wong
- Xixolog
- Buggy
- Sweet Acorn
- Peetan
- Jump Coaster
- Buggy 84
- 3D Water Driver
- Pinky Chase
- Wedding Bells
- Fightting Rider

Aplicación

- Memoria Ram 4 K
- Creative Greetings
- Character Collect
- Quinielas y Reducciones
- Pascal
- Ensamblador
- Generador Juegos

Gestión

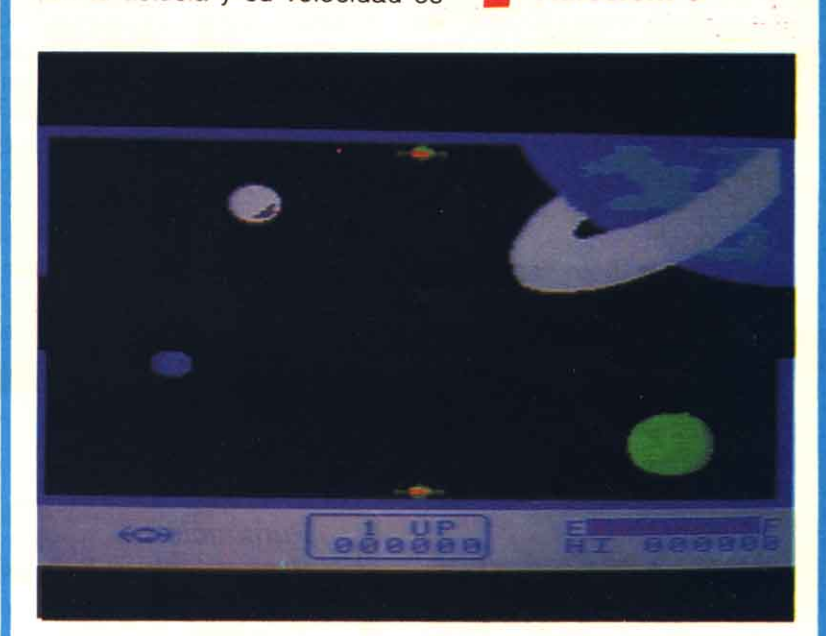
- Hoja de Cálculo
- Homewriter
- Control Stocks
- Contabilidad Personal
- Ficheros
- Procesador de Textos
- Control Stocks
- Vencimientos
- Contabilidad 1.500

SOFTWARE

un recorrido extraño y cronometrado, en el que tendrás que realizar un «reto», si lo consigues, se te dará una vida más, sino, no pasa nada.

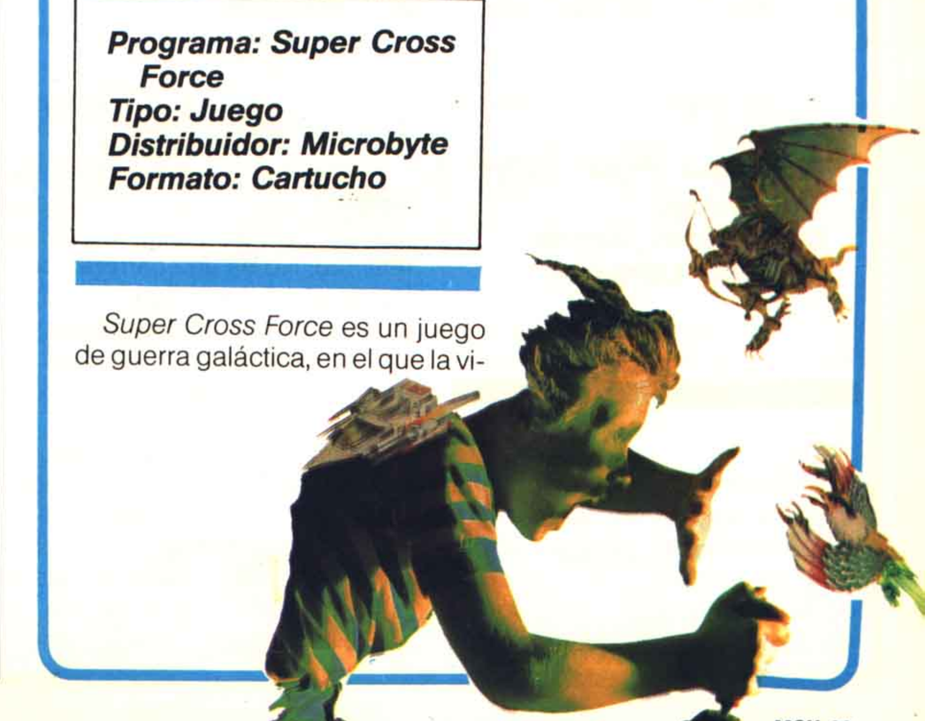
Es un juego muy divertido para todos, y sin duda es una gran aventura. Juega a ser *Rockford*, y con tu astucia y su velocidad os haréis millonarios y disfrutaréis, además de divertiros.

Puntuacion:
Presentación: 8
Claridad: 7
Rapidez: 7
Adicción: 9



Programa: Super Cross Force
Tipo: Juego
Distribuidor: Microbyte
Formato: Cartucho

Super Cross Force es un juego de guerra galáctica, en el que la vi-



da del planeta se ve acosada por los continuos ataques de la fuerza enemiga, los *Morpul*, invasores despiadados que nos quieren aniquilar, pero nuestras naves están capacitadas para defenderse, y no les dejarán que nos destruyan.

Contamos para ello con un sistema de disparo muy diferente al que nos ofrecen otros juegos. Dos naves situadas en la parte superior e inferior de la pantalla, crearán fuego cruzado en paralelo e incluso en zig-zag, pero todo depende de nuestra habilidad.

El juego consta de una pantalla única, donde desarrollaremos nuestra batalla, pero serán diversos tipos de naves las que nos ataquen. Aparecerán por cualquier lado de nuestra pantalla y a mucha velocidad, disparando misiles que tendremos que batir con mucha astucia, ya que si nos derriban las tres naves con las que contamos incialmente habrá acabado la partida.

El elemento principal del juego es tener la suficiente habilidad para no ser batidos y conseguir recoger el paracaidista que en cualquier momento nos será lanzado desde las alturas por una nave nodriza, y que si no recogemos quedará perdido en el espacio para siempre.

Tener un juego de estas características es siempre aconsejable, más si tenemos en cuenta el mal tiempo de estas fechas, por lo que pasaremos más de una tarde disfrutando de las batallas galácticas y de nuestra habilidad como «guerreros espaciales» sin salir de casa. Una última característica que hay que tener en cuenta, es que pueden jugar al mismo tiempo dos personas contra la máquina, lo que siempre es un aliciente.

En resumen, estamos ante un bonito juego de guerra, creado con detalle y con una muy depurada técnica de programación.

Puntuación:
Presentación: 8
Claridad: 8
Rapidez: 7
Adicción: 7

Programa: Hyper Rallye
Tipo: Juego
Distribuidos: Serma
Formato: Cartucho

Poder competir en todos los campos del deporte es cuestión de proponérselo, aunque esto sea muy difícil, como sucede en el deporte de la Fórmula 1, pero ¿quién ha dicho que no podamos tener una carrera propia y competir en nuestros propios hogares?

Hyper Rally es un juego creado para los amantes de la velocidad que no pueden hacer sus sueños realidad. No es una carrera como las demás; nos veremos metidos en un sin fin de complicaciones al ver a nuestros contrincantes, ya que no se trata de ganar, sino de llegar.

Estamos en la decimotercera prueba del Campeonato Mundial de Rallyes; la carrera consta de un total de trece etapas y todas ellas tienen lugar en circuitos potentes y famosos. ¡Adelante y aléjese de sus pilotos rivales! ¡Haga frente a las adversidades de la prueba!, y utilice su magnífica técnica y pericia que le llevarán a ser el primero en la línea de meta.

Es un juego diseñado para jugar una sola persona, bien puede ser utilizando joystick o cursores.



Comenzará pulsando la barra espaciadora para arrancar, el cursor superior le servirá de freno y el inferior para acelerar, aquí debe tener en cuenta que su coche consta de dos velocidades, es decir,



utiliza engranajes de alta y baja multiplicación. El cambio lo efectuará automáticamente el ordenador, pero será usted el que dirija a la máquina, no se olvide.

Comenzará la prueba en un lugar asignado por la máquina, y si al finalizar la misma se encuentra dentro de esa posición, pasará a la etapa siguiente.

Debe contar con los peligros que se le avecinan, ya que los despiadados conductores que van a competir contra usted le originarán muchas dificultades si no se anda con cuidado; por otra parte, su combustible no es ilimitado y si lo consume antes de finalizar la prueba, ésta habrá terminado para usted.

Es un juego sencillo que no tiene complicaciones, al que pueden jugar todos, sin excepción, disfrutando del colorido de los paisajes y del maravilloso ruido del motor, que le hará imaginar velocidades insospechadas y vivir todas las sensaciones que le puede dar ver una prueba de estas características.

¡Adelante y a ganar!

Puntuación:
Presentación: 8
Claridad: 7
Rapidez: 7
Adicción: 8

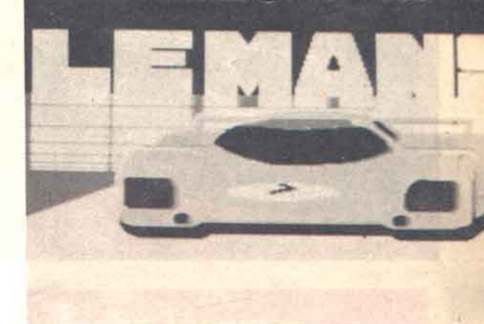
Programa: Le Mans
Tipo: Juego
Distribuidor: Dynadata
Formato: Cassette

Competir en el mundo de la Fórmula 1 ha sido uno de los principales temas de juego llevado al ordenador, pero muchos de estos simuladores de carreras no nos hacen experimentar todas las sensaciones que concurren en un deporte tan difícil y peligroso. *Le Mans* no es el juego de lucha y velocidad llevado al extremo de lo irremediable, sino que se nos presenta como la realidad convertida en juego.

Utilizar gráficos en tres dimensiones y una pantalla que nos señala el punto de vista del conductor, para obtener así mayor realidad, nos da la sensación de sumergirnos en lo que verdaderamente supone competir en la Fórmula 1.

Alejarnos del bullicio cotidiano y de los atascos es una ilusión que a menudo nos invade, pero que en muchos casos resulta irrealizable. Aquellos amantes de la conducción que ven frenadas sus ilusiones, ahora tienen la oportunidad de conseguirlas, disfrutando al mismo tiempo de la naturaleza y del riesgo.

De todos es conocido que *Le Mans* es uno de los mejores circuitos mundiales y en el han demostrado su pericia y habilidad grandes campeones de este deporte, ahora tenemos la posibilidad de ser nosotros uno de ellos. Nuestro coche no tendrá nada que envidiar a esas máquinas de precisión creadas para quemar los neumáticos en la carretera, ya que nuestro automóvil desarrollará una potencia de 360 km/h y será un modelo de competición del grupo G, diseñados para morir en la carretera.



El color, tan importante en el mundo del deporte, y especialmente en la Fórmula 1 donde se dan cita multitud de personas y elementos, ha sido conseguido al hacernos vivir la competición como espectadores de esta gran lucha por conseguir la victoria.

Los motores comienzan a rugir. La luz de salida se pone en verde. Suelta el embrague entre el bramido y el olor a goma quemada... ¡comienza la carrea!

Puntuación:
Presentación: 9
Claridad: 8
Rapidez: 7
Adicción: 7





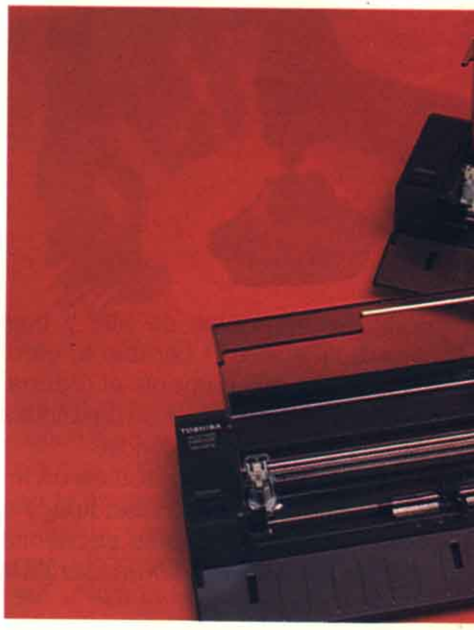
Plotters para MSX

Dentro de la amplia gama de periféricos disponibles para los ordenadores MSX, las impresoras forman un grupo numeroso y diverso. Sin embargo, dentro de esta variedad, existen unos aparatos cuya labor no sólo se limita a imprimir los listados de los programas; estamos hablando de los plotters.

Estos se presentan como la opción más completa e interesante del mercado, sobre todo si nuestro trabajo requiere el uso de dicho periférico. Dos son los que traemos a nuestras páginas, el PRN-C41 de SONY y el HX-P570 de TOSHIBA.

Lo que mas nos sorprendió de ellos fue su reducido tamaño y su extraña similitud, aunque ambos ofrecen unas prestaciones más que aceptables para la mayoría de los usuarios.

De cualquier manera, tienen un inconveniente que a la larga es determinante, la lentitud con que imprime programas. Esto se debe a que los caracteres son dibujados por una pluma, tarea laboriosa hasta para una máquina. Sin embargo, debido a que la impresión se efectúa con 4 plumas, se obtiene una gran calidad de escritura con la posibilidad de escribir en 15 tamaños diferentes y en 4 colores: verde, azul, rojo y negro.



Una vez seleccionado el modo gráfico, se convierte en un magnífico plotter con el que podemos realizar cualquier dibujo que nos propongamos. Además, existen una serie de comandos que podemos utilizar una vez seleccionado dicho modo. Estas instrucciones controlan todos los movimientos

de las plumas, a la vez que se determina la dirección en la que tiene que ir el papel. Por ejemplo, podemos mover la pluma a unas coordenadas específicas, cambiar el color de ésta, dirigir la pluma (o el papel) a una posición relativa, dibujar una recta entre dos puntos o escribir una cadena de caracteres. De cualquier modo, además de poseer estas funciones, imprescindibles en cualquier plotter, disponemos de otros comandos menos importantes pero



***Foto 1:** La similitud entre los dos aparatos dificulta su elección. Salvo el logotipo, lo demás es idéntico.*

que resultan igualmente útiles a la hora de realizar un dibujo. Existen instrucciones que permiten, por ejemplo, seleccionar el tamaño de los caracteres, imprimir caracteres girados, dibujar líneas discontinuas, etc. En cuanto al papel utilizado para imprimir, podremos optar por utilizar el rollo que se suministra con el aparato u hojas sueltas tamaño folio.

Con el plotter, también se suministra además del rollo de papel y el portarrollos correspondiente, la fuente de alimentación para su conexión a la red, el cable interface para el conector Centronics del ordenador, las cuatro plumas de dibujo, una cinta con un programa de demostración y los manuales correspondientes.

En él, se explica claramente la conexión, puesta en marcha y mantenimiento del equipo, así co-



***Foto 2:** Vista del conector Centronics y de los switches que cotrolan las distintas opciones del aparato.*

TITL	A	B	C	D	MEAN	TOTAL
ENE	100	100	0	0	100	200
FEB	125	127	0	0	126	252
MAR	113	200	0	0	156	313
ABR	120	120	0	0	120	240
MAY	260	340	0	0	300	600
JUN	390	400	0	0	395	790
JUL	400	410	0	0	405	810
AGO	420	420	0	0	420	840
SET	500	430	0	0	465	930
OCT	600	440	0	0	520	1.040
NOV	400	500	0	0	450	900
DIC	360	600	0	0	480	960
MEAN	315	340	0	0	328	656
SUM	3.788	4.087	0	0	3.937	7.875

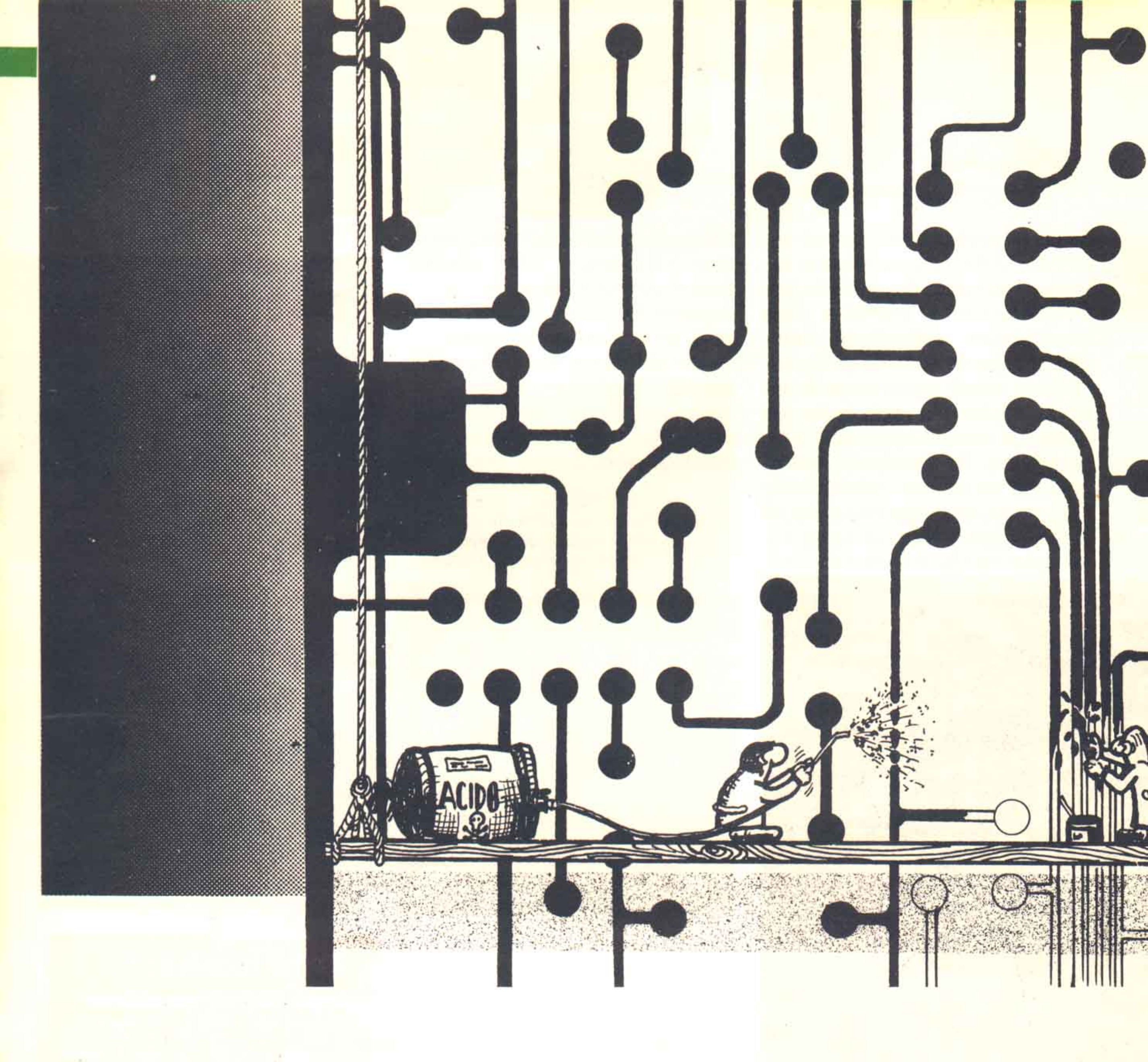
Ejemplo de una aplicación con plotter.

mo las instrucciones para su control incluyendo un pequeño programa de ejemplo. El programa suministrado con el equipo permite representar gráficamente una serie de datos. Dispone de gran cantidad de opciones que permiten introducir y modificar datos, realizar gráficos de barras, añadir comentarios, salvar y cargar datos, etc. Este programa viene acompañado de un extenso manual que facilita su manejo por parte del usuario, lo que se traduce en una facilidad de uso y en un aprovechamiento máximo de las posibilidades gráficas tanto del ordenador como del plotter.

Es una alternativa más a la hora de comprar una impresora con la ventaja adicional de las posibilidades gráficas, a un precio más asequible que muchas de las impresoras convencionales.

Ficha técnica:

Tensión de alimentación: 9,8 V.
Consumo de potencia: 12 W.
Dimensiones: 310x67x108 mm. (ancho/alto/fondo).
Sistema de impresión: 4 Plumas giratorias.
Velocidad de dibujo: X,Y 57 mm/seg 81 mm/seg en ángulo de 45 grados.
Velocidad de impresión: 6 CPS.
Resolución: Paso de 0,2 mm.
Número de caracteres: 255 con la posibilidad de imprimirlos en 16 tamaños.
Caracteres por línea: 160 (tamaño pequeño de letra).
Colores: Negro, Azul, Verde y Rojo.
Códigos: Estándar MSX.



Más memoria utilizable

El programa se encarga de utilizar las otras 32k de memoria RAM que no son útiles desde BASIC, como una memoria secundaria donde guardar un programa mientras se trabaja con otro. Esto nos permite trabajar en dos programas distintos e independientes, sin tener que traer y guardar programas en cassette, con el consiguiente ahorro de tiempo.

El programa viene en BASIC. Cuando se pone en marcha, reserva memoria para el C/M y lo pasa de la de los DATAS a la me-

moria, inicializa las tres primeras teclas de función y se borra permitiendo al usuario trabajar normalmente.

Las tres teclas de función se inicializan de la siguiente forma:
F1: guardar el programa
F2: traer el programa
F3: intercambiar los programas

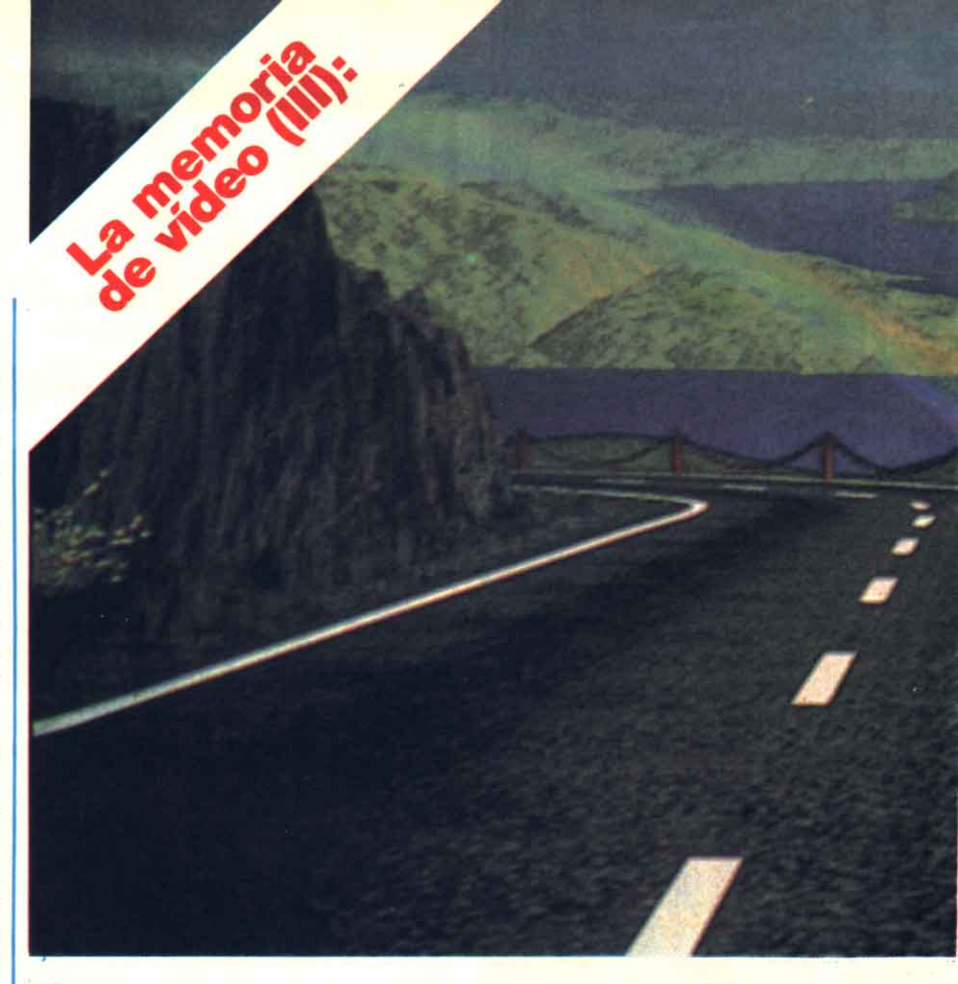
El programa en BASIC no tiene mayor secreto, por lo que describiremos el programa en ensamblador.

Está dividido en cuatro partes, INICIO, ESCRIB, LEER e INTER. La primera parte se encarga, en cuanto se pone en marcha el programa cargador, de averiguar dóndo se encuentran las otras 32k RAM, y guarda este dato en la posición de la ROM y RAM usadas por el sistema. La posición DATO se utiliza para indicarle al BASIC si no se han encontrado las 32k buscadas. Si esta posición mantiene un 255, es que todo a ocurrido correctamente, si es cero ha habido algún error. El error puede ser porque la posición de la puerta A del chip 8255 PPI no esté en la dirección &HA8. La segunda parte es la encargada de guardar un programa y todas sus variables (incluídas las variables del sistema) en las 32k RAM no utilizadas por el BASIC. La tercera se encarga de traer un programa previamente guardado. La cuarta se encarga de intercambiar un programa guardado con el programa que tengamos en la memoria principal.

El programa se guarda entre las direcciones &HFØØØ y &HFØAD, lo que deja mucha RAM libre para el BASIC, pero impide utilizar una unidad de discos.

Juan D. Sandoval González
Madrid

```
10 XX=Ø:CLEAR 2ØØ.&HEFFF:RESTORE 3ØØ
20 FOR N=&HFØØØ TO &HFØA7
3Ø READ A$:A=VAL("&H"+A$):XX=XX+A:POKE N.A
4Ø NEXT N
45 IF XX<>22979 THEN PRINT"ERROR en la entrada de datos":PRINT:PRINT"     revisel
os":END
5Ø DEFUSR=&HFØØØ:A=USR(Ø)
6Ø DEFUSR1=&HFØ4A:DEFUSR2=&HFØ62:DEFUSR3=&HFØ7A
7Ø KEY1."?USR1(Ø)"+CHR$(13)
8Ø KEY2."?USR2(Ø)"+CHR$(13)
9Ø KEY3."?USR3(Ø)"+CHR$(13)
1ØØ CLS:PRINT:PRINT
11Ø PRINT"     F1.........GUARDAR"
12Ø PRINT"     F2..........CARGAR"
13Ø PRINT"     F3....INTERCAMBIAR"
14Ø NEW
3ØØ DATA F3.ØE.A8.ED.78.32.AA.FØ,Ø6.ØF.E6.FØ.BØ.57.E6,Ø3.7A.28.ØB.ED.79.57.C5.CD
.32.FØ.C1;7A.28.Ø9.1Ø.EA.AF.32.AB.FØ.C3.A1.FØ.32.AC.FØ.3E.FF.32.AB.FØ.C3.A1.FØ
31Ø DATA 21.FE.3F.3E.AA.Ø6.Ø4.77.23.1Ø.FC.21.FE.3F.Ø6.Ø4.7E.23.FE.AA.CØ.1Ø.F9.C9
.F3.ED.73.A8.FØ.3A.AC.FØ.D3.A8.11.ØØ.ØØ.21.ØØ.8Ø.Ø1.ØØ.8Ø.ED.BØ.C3.A1.FØ.F3
32Ø DATA 3A.AC.FØ.D3.A8.11.ØØ.8Ø.21.ØØ.ØØ.Ø1.ØØ;8Ø.ED.BØ.ED.7B.A8.FØ.C3.A1.FØ.F3
.ED.73.A8.FØ.3A.AC.FØ.D3.A8.11.ØØ.ØØ.21.ØØ.8Ø.Ø1.FF.7F.1A.32.FF.FF.ED.AØ.2B
33Ø DATA 3A.FF.FF.77.23.78.B1.2Ø.FØ.ED.7B.A8.FØ.3A.AA.FØ.D3.A8.FB.C9
```



Los secretos

El modo de pantalla de alta resolución, SCREEN 2, tiene muchas aplicaciones y características cuyo conocimiento permitirá utilizar las posibilidades del MSX hasta el límite, aprovechando cada byte de memoria utilizable.

El modo de gráficos *SCREEN 2* es el más adecuado para efectuar los dibujos que necesitéis. Podéis emplear libremente cualquiera de los 256 x 192 pixels de la pantalla con unas mínimas limitaciones de color.

Los modos de gráficos son una de las mayores ventajas que posee el *BASIC MSX,* el cual, gracias a su *GML* (Graphics Macro Language = Macrolenguaje para Gráficos) puede lograr unos dibujos extremadamente complejos en pocos segundos. El *GML* se usa exclusivamente con la instrucción *«DRAW»,* que se asemeja al lenguaje *LOGO,* tradidionalmente utilizado para el diseño de gráficos basado en instrucciones muy sencillas. Gracial al *GML* se puede orientar un lápiz imaginario en 8 direcciones distintas y, a continuación, pintar o borrar trazos en la pantallas. Esas 8 direcciones son las cuatro típicas (Arriba, Abajo, Derecha e Izquierda) y las 4 direcciones diagonales. A través del *GML* también es posible determinar un movimiento relativo o vectorial, un movimiento absoluto, el color en que ha de pintarse un trazo. Es posible también girar los ejes coordenados, y hasta cambiar la escala de dibujo.

Pero el *GML* y todas estas características que acabamos de resumir, es únicamente *BASIC,* y nuestro objetivo es llegar más allá, en suma, dominar la pantalla desde la memoria de vídeo. Por supuesto que para emplear la memoria de vídeo se usan instrucciones en *BASIC (VPOKE y VPEEK* son mandatos en *BASIC),* pero la memoria de vídeo no es estrictamente de *BASIC:* se utiliza por igual desde *BASIC* que desde Código Máquina. En realidad constituye un paso más en el conocimiento de nuestro ordenador.

Como más adelante veréis, gracias a la memoria de vídeo podéis prescindir de *PSET* y *PRESET,* pero no os aconsejamos dejar el *GML* ni las instrucciones *CIRCLE* y *LINE.* Estas tres cosas son muy útiles, y resultan insustituibles en la memoria de vídeo, excepto en dibujos pequeños.

La tabla 12: El mapa de la pantalla

La tabla 12 es, efectivamente, un mapa de la pantalla, pero un mapa muy especial. Si vosotros tenéis el mapa de un país, con cabos, golfos o bahías y decidís borrar del mapa un cabo, por ejemplo, lo único que tendréis será un mapa incorrecto, pues ese cabo sigue existiendo en la realidad, a pesar de que en vuestro mapa lo hayáis borrado. Sin embargo, si hay una serie de *pixels* pintados en vuestro «mapa de la pantalla», es decir, en la tabla 12, y borráis algunos, inmediatamente la pantalla los borrará, y lo mismo sucederá si en vez de borrarlos, los pintarais.

La tabla 12 es, junto a la tabla 11, que más adelante explicaremos, la más grande de todos los sistemas de almacenamiento que puede adoptar la memoria de vídeo. Tiene una longitud de 6144



de SCREEN 2

octetos, y en nuestros ordenadores MSX comienza en el octeto 0, terminando en el 6143.

Los 256 x 192 *pixels* que tiene la pantalla se agrupan en cuadros de 1 fila x 8 columnas de *pixels*. Cada cuadro tiene, por lo tanto, ocho *pixels*. Debido a que estos cuadros tienen una forma de rectángulos muy alargados, a menudo se les llama «rebanadas» de *pixel* (nosotros lo llamaremos agrupación). Cada agrupación se corresponde con un octeto de la tabla 12, de tal manera que cada uno de los 8 *pixels* se corresponde con cada una de las 8 cifras en binario que tiene el número que contiene su octeto. Estos, se reúnen en grupos de ocho, formando una «torre» (V. Figura 1) por lo tanto cada torre posee 8 filas x 8 columnas de *pixels*. El tamaño de una torre coincide con el de los cuadros del modo *SCREEN 1*.

Así funciona la tabla 12: los octetos del 0 al 7, ambos inclusive, se corresponden con los 8 *bytes* que constituyen la primera torre, situada en la esquina superior izquierda de la pantalla. Dentro de esa torre, el *byte* superior se corresponde con el octeto 0, el inferior con el 1, el siguiente con el 2,... y el *byte* inferior con el 7. (V. Figura 2). El octeto 8 se corresponde con el *byte* superior de la segunda torre, que se encuentra a la derecha de la primera. Lógicamente, el *byte* inferior de la segunda torre se corresponderá con el octeto 15. La tercera torre se sitúa también a la derecha de la segunda, y la cuarta a la derecha de la tercera, etc. A la derecha de la trigesimoprimera torre se halla la trigesimosegunda, que es la torre de la esquina superior derecha, cuyo *byte* inferior se corresponde con el octeto 255.

El GML o Graphic Macro languaje, similar al LOGO, permite realizar cualquier tipo de diseño.

El octeto 256 se corresponde con el *byte* superior de la primera torre de la segunda fila de torres, que se halla inmediatamente debajo de la esquina superior izquierda. En toda la pantalla existen 24 filas de 32 torres cada una. Si cada torre consta de 8 agrupaciones, y cada una de éstas es un octeto, necesitaremos 6144 octetos para todas las agrupaciones (32 x 24 x 8 = 6144). Por eso la agrupación inferior de la última torre, la torre de la esquina inferior derecha, se corresponde con el octeto 6143.

Queda claro, por tanto, que cada *pixel* se corresponde con una cifra en binario de un octeto, cada agrupación con un octeto y cada torre con 8 octetos. Como en algunas de las tablas explicadas en números anteriores, si la cifra que se corresponde con un *pixel* es «0», éste forma el color del fondo, y si es «1» lo toma del color de la tinta.

Tras una deducción lógica que no incluimos aquí por ser demasiado complicada, hemos averiguado la fórmula de la cifra en binario correspondiente a un *pixel* de la pantalla en función de su fila y columna. Hemos considerado que la primera cifra en binario es la cifra de la izquierda. La octava cifra será, por lo tanto, la última cifra a la derecha.

Si queréis rellenar de color de tinta un *pixel* tenéis que convertir en «1» la cifra número:

n.º de columna MOD 8 + 1 del octeto número primer octeto de la tabla + INT (n.º de fila/8) x 256 + INT (n.º de columna/8) x 8 + n.º de fila MOD 8.

Si queremos, por ejemplo, rellenar el *pixel* (195,73) de color de la tinta será necesario convertir en «1» la cifra cuarta (195 MOD 8 + 1 = 4) del octeto 2497 (0 + 2304 + 192 + 1 = 2.497).

La tabla 11: Los colores en SCREEN 2

Si la tabla 12 consistía en un mapa de la pantalla, la tabla 11 completa ese mapa, indicando qué colores ha de tomar cada agrupación. La tabla 11 consta de 6144 octetos, como la tabla 12, y cada octeto se corresponde con una agrupación. Así se cubren las 6144 agrupaciones de la pantalla. En nuestros ordenadores MSX la tabla 11 comienza en el octeto 8192, terminando en la 14335.

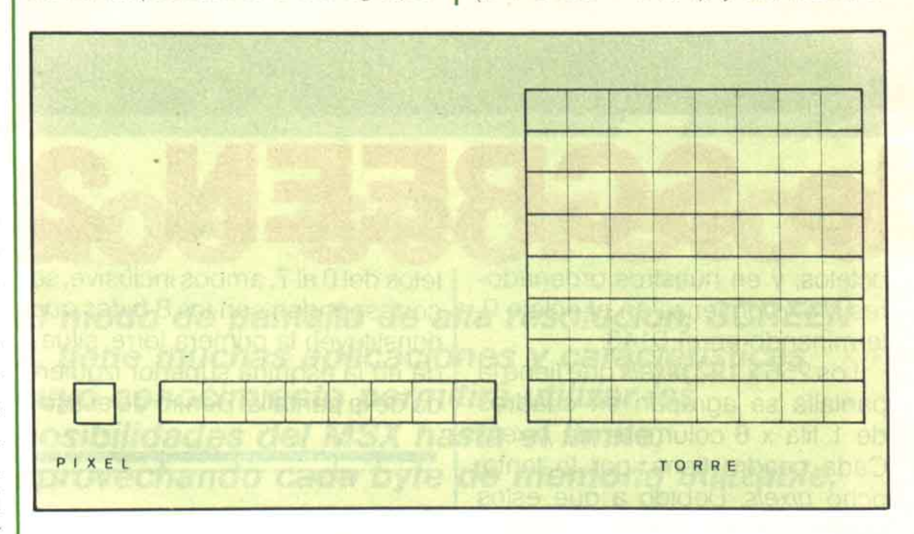
Al igual que era conveniente tratar los datos de la tabla 12 en sistema binario, en la tabla 12 conviene tratarlos en sistema hexadecimal. La primera cifra hexadecimal indica el color de tina, y la segunda el color de fondo que debe adquirir la agrupación a la que corresponda ese octeto. (La lista con los colores y sus valores hexadecimales la publicamos en el número anterior.)

Vamos a aclarar esto un poco más. Si vosotros mandáis a vuestro ordenador *MSX* que se sitúe en *SCREEN 2* con color de fondo blanco y cualquier color de tinta, todos y cada uno de los 6144 octetos de la tabla 11 introducen instantáneamente el número &H0F, de tal manera que aquellos *pixels* a los que en la tabla 12 corresponda un «0», queden coloreados, y aquellos a los que corresponda un «1», tomen el color de la tinta que hay que especificar en la tabla 11, o bien desde una orden en *BASIC* para gráficos. Pero si introducís después en un octeto otra combinación de colores, por ejemplo el número &H81 (tinta roja sobre fondo negro), únicamente obedecerán esa combinación los *pixels* pertenecientes a la agrupación correspondiente a ese octeto, mientras que las demás agrupaciones de la pantalla permanecerán «leales» al color que establecimos al principio.

Vayamos ahora con los detalles técnicos: los octetos de la tabla 11 se corresponden con sus agrupaciones de la misma manera que los de la tabla 12 con las suyas, es decir, el octeto 8192 corresponde a la agrupación superior de la torre de la esquina superior izquierda, el octeto 8193 a la de debajo, el 8194 a la de debajo, y así hasta el 8199, que corresponde a la agrupación inferior. El 8200 corresponde a la agrupación superior de la torre situada a la derecha de la primera, etc. Por último, el octeto 14335 corresponde a la agrupación inferior de la torre de la esquina inferior derecha.

Hay dos maneras de encontrar el octeto correspondiente a una agrupación. La primera es la típica: si se toma el número de fila y columna de un *pixel*, el octeto que determina su color y el de los otros *pixels* de su misma agrupación es: primer octeto de la tabla + INT (n.º de fila/8) x 256 + INT (n.º de columna/8) x 8 + n.º de fila MOD 8.

Podéis observar que la fórmula es la misma que la de la tabla 12, y por eso hay otra manera de calcularlo. Si el octeto 0 rige la misma agrupación que es 8192 (0 + 8192 = 8192), y el octeto 1 de la misma que el 8193 (1 + 8192 = 8193), y el 2 la misma que el 8194 (2 + 8192 = 8194), y así hasta el 6143, que rige la misma agrupación que el 14335 (6143 + 8192 = 14335), está claro que si de una agrupación ya tenemos su octeto en la tabla 12, al sumar a este octeto el número 8192, hallaremos el octeto de la tabla 11 que se ocupa de la misma agrupación.



Las tablas 11 y 12 en la práctica

Vamos a comenzar por las aplicaciones casi exclusivamente de la tabla 12. Por ejemplo, al *pixel*

(195,73) al que antes hacíamos referencia, vamos a colorearlo de azul oscuro:

```
10 COLOR, 4 15,15:SCREEN 2
20 VPOKE 2497 + 8192, &H4F:
   VPOKE 2497,&00010000
30 GOTO 30
```

El programa anterior hace exactamente lo mismo que éste:

```
10 COLOR 4, 15, 15:SCREEN 2
20 PSET ( 195,73 )
30 GOTO 30
```

En este ejemplo, la instrucción *PSET* es igual de útil que la tabla 12, pero en el caso de que sean varios *pixels* de una misma agrupación, conviene usar la tabla 12.

Esta tabla es utilísima para dibujos complicados, los cuales han de pasar antes de ser programados por un proceso de «digitalización». La digitalización consiste en determinar cuáles son las coordenadas de todos los *pixels* que intervienen en un dibujo. Una vez digitalizado el dibujo, los tados de sus *pixels* se introducen en la tabla 12 por medio de mandatos *VPOKE*, y el color de cada *pixel* se introduce también en la tabla 11 de la misma manera. Tras esto, el dibujo está terminado. Si los datos a introducir son muchos y queréis ahorrar memoria, podéis transformar los datos en binario a sistema hexadecimal, y los números que antes ocupaban 8 caracteres (como el número «11010100», por ejemplo) ahora ocupan sólo 2 (como «D4»).

También es posible, utilizando solamente esta tabla, lograr el efecto *«FLASH»*. Vamos a demostrártelo con un ejemplo: digitalicemos dos rombos concéntricos que ocupen una torre de agrupaciones:

&B00011000 = &H18 = 24
&B00100100 = &H24 = 36
&B01011010 = &H5A = 90
&B10100101 = &HA5 = 165
&B10100101 = &HA5 = 165
&B01011010 = &H5A = 90
&B00100100 = &H24 = 36
&B00011000 = &H18 = 24

Hallemos a continuación sus opuestos, aquellos números binarios que poseen «1» donde los números anteriores tienen ceros, y poseen ceros donde los números anteriores tienen «1»:

&B11100111 = &HE7
&B11011011 = &HDB
&B10100101 = &HA5
&B01011010 = 5H5A
&B01011010 = 5H5A
&B10100101 = 5HA5
&B11011011 = 5HDB
&B11100111 = &HE7

Con una representación de 256 x 192 pixels, los gráficos conseguidos son muy realistas y detallados.

Estos números binarios forman otro dibujo: el negativo del doble rombo. Alternando estos dos dibujos se logra el efecto *«FLASH»*.

Pero la digitalización de ambos aún no está terminada. Falta lo más importante: saber en qué lugar de la pantalla van a aparecer estos dibujos. Vamos a situarlos, por ejemplo, en una torre enclavada en el centro de la pantalla. Nos referimos a la torre regida por los octetos de 2944 al 2951. El programa sería el siguiente:




```
10 COLOR 4,15,15:SCREEN 2:
   FOR T=11136 TO 11143
   :VPOKE T,&H4F :NEXT
20 FOR T=2944 TO 2951:READ
   A$:VPOKE T,VAL ("&H" + A$):
   NEXT
30 FOR R=1 TO 300:NEXT
40 FOR T=2944 TO 2951:READ
   Q$:VPOKE T,VAL("&H" + Q$):
   NEXT
50 FOR R=1 TO 300:NEX: RES-
   TORE: GOTO 20
60 DATA 18,24,5A,A5,A5,5A,24,
   18
70 DATA E7,DB,A5,5A,5A,A5,DB,
   E7
```

Se incluyen en el programa dos bucles de variable R para retardar el cambio de negativo a positivo o viceversa. Los octetos del 11136 al 111143 son los que corresponden a esa torre en la que se alternan los dos dibujos.

Como podéis observar, las líneas 20 y 40 son iguales, pues ambas realizan la misma función: recoger los datos de los rombos,

transformarlos de sistema hexadecimal a sistema decimal y situarlos en la tabla 12.

Pasemos ahora a las aplicaciones de la tabla 11, que es la de mayor permisividad en materia de colores que podéis encontrar en cualquier modo *SCREEN* de vuestro ordenador MSX. Al caber dos colores por agrupación, en una torre pueden caber hasta 16 colores distintos. Con tal peculiaridad, el efecto «*FLASH*» puede tomar una espectacularidad insospechada; por ejemplo, con los dos rombos de antes:

```
10 COLOR 15:SCREEN 2
20 FOR T=2944 TO 2951:READ
   Q$:VPOKE T,VAL ("&H" + Q$):
   NEXT
30 DATA  18,24,5A,A5,A5,5A,24,
   18
40 FOR T=2 TO 9:VPOKE 11134
   + T,240 + T:NEXT
50 FOR R=1 TO 300:NEXT
60 FOR T=2 TO 9:VPOKE 11134
   + T,T x 16 + 15:NEXT
70 FOR R=1 TO 300:NEXT:GOTO
   40
```

En este programa se realiza el efecto «*FLASH*» con un haz de 8 líneas horizontales que revisten el doble rombo. En la línea 20 del programa se emplea la tabla 12 para dibujar los dos rombos concéntricos, basándose en los datos suministrados en la línea 30, que son los mismos que calculamos antes. Si el doble rombo se sitúa en la torre cuya agrupación superior está regida por el octeto 2944 en la tabla 12, en la tabla 11 estará regida por el octeto 11136 (2944 + 8192 = 11136).

Para elegir los 8 colores de fondo, hemos tomado la lista de colores MSX y hemos seleccionado los colores del 2 al 9. Hemos establecido también el bucle de variable T de las líneas 40 y 60 en función de estos colores. Para que el primer color (el 2) coincida con el primer octeto de la torre (el 11136), disponemos en el mandato *VPOKE* que el color sea el «11134 + T», pues si es el color 2 será el octeto 11136, si es el 3 será el 11137, si es el 4 será el 11138, etc. En la línea 40 se introducen estos números en la tabla 11, sumándoles 240 (240 es &HF0, lo que determina un color de tinta blanco) y representando estos números el color de fondo. Sin embargo, al multiplicar en la línea 60 el color por 16, la primera cifra hexadecimal (el color de tinta) será la que dependa del octeto, y el color de fondo en todos será blanco (15 es &H0F en hexadecimal).

La tabla 10: Un secreto muy bien guardado

La tabla 10 es, posiblemente la más útil de todas las de la memoria de vídeo. Lo que esta pequeña tabla (768 octetos de longitud) puede hacer es inimaginable desde el *BASIC*. Su poder supera al de las otras tablas de *SCREEN 2*, y no hay ningún elemento en *BASIC* que nos pueda hacer sospechar su existencia. la tabla 10 es, como decíamos, el secreto mejor guardado de la memoria de vídeo.

La tabla 10, con sus 768 bytes, es muy útil, superando ampliamente otras tablas de mayor capacidad.

Como ya quedó claro antes, existen en la pantalla 6144 agrupaciones, que se agrupan de 8 en 8 formando un total de 768 torres. A cada una de estas torres le corresponde un octeto de la tabla 10, y aquí viene lo más interesante: cada octeto le indica a su torre correspondiente en función de qué torre debe pintarse.

Vamos a aclarar este asunto. Una torre de agrupaciones se pinta siempre según le indican los 8 octetos de sus agrupaciones en la tabla 12, y de los colores que le indican los ocho octetos de sus agrupaciones en la tabla 11; estos dos hechos, que hasta ahora eran normales, sólo suceden si el octeto que corresponde a esa torre en



la tabla 10 lo permite: sólo suceden si esa torre está en función de sí misma. Pongamos un ejemplo, la torre número 0, a la que antes llamábamos «primera torre», que se halla en la esquina superior izquierda, adopta los patrones que le marcan sus octetos de la tabla 12 (octetos del 0 al 7, ambos inclusive) y obedece los colores que le ordenan sus octetos de la tabla 11 (octetos del 8192 al 8199, ambos

inclusive) en el caso de que esté en función de sí misma, es decir, en el caso de que su octeto de la tabla 10 tenga el número «0», pues es la torre número 0. Pero si su octeto de la tabla 10 tuviera el





número «1», esta torre actuaría como si fuera la torre número 1 (que es la torre que se halla inmediatamente a su derecha, a la que antes llamábamos «segunda torre»), es decir, adopta los patrones de la tabla 12 que corresponden a la torre 1 (octetos del 8 al 15, ambos inclusive), y obedece los colores de la tabla 11 que corresponden a la torre 1 (octetos del 8200 al 8207, ambos inclusive), todo ello sin dejar de ser la torre 0.

La tabla 10 comienza en el octeto 6144 y termina en el 6911. El octeto 6144 corresponde a la torre 0, el 6145 a la torre 1, el 6146 a la torre 2, y sigue en esta progresión hasta el 6175, que corresponde a la torre 31, la de la esquina superior derecha. La torre 32, correspondiente al octeto 6176, es la que antes denominábamos «primera torre de la segunda fila de torres» y está situada inmediatamente debajo de la torre 0. Así sigue esta sucesión, fila por fila, hasta la torre 255, la última de la octava fila de torres. La torre 255, para estar en función de sí misma, debe introducir en el octeto 399 el número 255. Pero como los octetos no admiten números decimales mayores de 255, la torre 256 es considerada como la torre número 0 del segundo «campo» de la pantalla, perteneciendo todas las torres anteriores al primero de estos «campos». Queda de esta manera bautizada la palabra «campo» como «zona de la pantalla donde caben 256 torres». La torre 257 será la torre 1 del segundo campo, la 258 la torre 2, y continuando llegaríamos a la torre 511, que es la torre 255 del segundo campo, que correspondería al octeto 6655. La torre 512 es la torre 0 del tercer y último campo de la pantalla, siendo la torre 255 de este tercer campo la torre 767. Ya no hay más torres; la torre 767 es la última, la de la esquina inferior derecha, y corresponde al último octeto, al 6911. Si este octeto contiene el número 255, la torre 767 está en función de sí misma.

Las tablas 11 y 12 son las más grandes de todas las que hay, ocupan 6144 bytes cada una.

La fórmula correspondiente a un octeto de la tabla 10 en función del n.º de fila de de columna de su torre es:

primer octeto de la tabla + n.º de fila x 32 + n.º de columna
y esa torre es la número:
(n.º de fila MOD 8) x 32 + n.º de columna
del campo número:
INT (n.º de fila / 8) + 1

NOTA: Al hablar de n.º de fila y n.º de columna nos referimos a filas y columnas de torres, no de *pixels.*

Por ejemplo, la torre 10 es la torre (0,10), y se corresponde con el octeto 6154 (6144+0x32+10= 6154), y es la torre número 10 ((0 MOD 8) x 32 + 10 = 10) del campo primero (INT (0/8) + 1 = 1).

Por esa razón, para poner la torre 10 en función de sí misma sería necesario introducir el número 10 en el octeto 6154.

La tabla 10 en la práctica

Vamos a probar los efectos de la tabla 10 en la pantalla del ordenador con las torres de la 0 a la 31 del campo primero. En primer lugar, cada una va a estar todas en función de sí mismas, y a continuación van a estar todas en función de la torre 31. Para que diferenciéis bien ambas partes del programa, en la primera el borde de la pantalla es negro, y en la segunda blanco. Vamos a determinar al azar los colores y los patrones de las torres de la 0 a la 30,

gracias a las tablas 11 y 12, mientras que la torre 31 va a tener sólo una franja roja para diferenciarla de las demás. Este el el programa:

```
10 COLOR ,4,1:SCREEN 2
20 FOR T=0 TO 247:VPOKE T,INT
   (RND(1) x 256):NEXT
30 FOR T=8192 TO 8439:VPOKE
   T,INT(RND(1) x 256):NEXT
40 VPOKE 8443,8
50 FOR R=1 TO 1000:NEXT
60 COLOR ,,15:FOR T=6144 TO
   6174:VPOKE T,31:NEXT
70 GOTO 70
```

de la 14: una fracción de circunferencia. Es un buen momento para distinguir bien los campos de la pantalla: toda la zona de fracciones de circunferencia forma el segundo campo.

Al ponerse todas estas torres en función de la 14, la circunferencia ha desaparecido, pues las torres que la constituían ahora no pueden adoptar sus patrones (que les indicarian dibujar la circunferencia completa) sino que han de adoptar el patrón de la 14. Sólo hay una excepción: la fracción de circunferencia que pasa por el primer campo de la pantalla no se ve influida por todo lo que sucede en el segundo campo, luego se puede seguir viendo perfectamente.

Este es el programa:

```
10 COLOR 15,4,4:SCREEN 2
20 CIRCLE (130,90),30,,,,1.4
30 FOR R=1 TO 500:NEXT
40 FOR T=6400 TO 6655:VPOKE
   T,14:NEXT
40 GOTO 50
```

ZONA DE BORDE

ZONA DE BORDE

OCTETO

0	8	16
1	9	17
2	10	18
3	11	19
4	12	20
5	13	21
6	14	22
7	15	23
256	264	
257	265	
258	266	
259	267	
260	268	
261	269	
262	270	
263	271	
512		
513		
514		
515		
516		
517		
518		
519		

Introducimos en la orden 50 un bucle retardador para que podáis observar detenidamente la primera etapa del programa.

El programa siguiente trata un efecto gráfico muy sencillo y al mismo tiempo muy interesante. En la primera etapa, el programa traza un círculo que podéis observar gracias al bucle ralentizador de la linea 30. A continuación sitúa todas las torres del segundo campo en función de la torre número 14,, torre por la que pasa la circunferencia antes trazada. Por esta causa todas las torres que están en función de la 14 adoptan el patrón

El último programa está destinado a que distingáis con claridad los tres campos de la pantalla. En primer lugar determina al azar, con la tabla 11, los colores de las agrupaciones de tres torres en la pantalla. Esas torres son las número 31 del primer, segundo y tercer campos. Un bucle retardador muestra estos colores para que los podáis observar bien. Después introduce a todos los octetos de la tabla 10 el número 31, lo que trae diversas consecuencias: las torres del campo primero han de ponerse en función de la torre 31 del campo primero, las del campo segundo en función de la 31 del campo segundo, y lo mismo sucede con las del campo tercero. Al tomar distintos colores, podéis observar la extensión de los tres campos de pantalla:

```
10 COLOR 14,4,4:SCREEN 2
20 FORT T=8440 TO 8447:VPO-
   KET, INT(RND(1) x 246):NEXT
30 FOR T=10488 TO
   10495:VPOKE T,INT (RND (1)
   x 256):NEXT
40 FOR T=12536 TO
   12543:VPOKE T,INT(RND(1) x
   256):NEXT
50 FOR R=1 TO 800:NEXT
60 FOR T=6144 TO 6911:VPOKE
   T,31:NEXT
70 GOTO 70
```

Las figuras móviles

Hay otras dos tablas en el sistema de almacenamiento del modo *SCREEN* 2, la 13 y la 14, pero tienen como misión ocuparse de las figuras móviles, por lo que serán tratadas en el número que destinemos a dichas figuras.

En el próximo número hablaremos de las tablas del sistema de almacenamiento de datos del modo *SCREEN* 3, y de las peculiaridades de este modo de gráficos de baja definición.

José María Cavanillas

SERVICIO DE EJEMPLARES ATRASADOS

ESTOS SON LOS EJEMPLARES DE MSX MAGAZINE APARECIDOS EN EL MERCADO CON UN RESUMEN DE SU CONTENIDO



Núm. 1
¿Qué es el MSX? Su BASIC, periféricos, programas, software.



Núm. 2
Generación de sonido. MSX-DOS, el ordenador por dentro, programas, noticias.



Núm. 3
Los joysticks, 256 caracteres programables, Z80 corazón de león, compro/vendo/cambio.



Núm. 4
Las comunicaciones entre ordenadores, la jerga informática, trucos, rincón del lector.

Núm. 5
Comandos de entrada/salida, el BASIC MSX comparado con Spectrum y Commodore 64. Código Máquina.



Núm. 6
Los 8 magnificos (test gigante), el bus de expansión, los misterios de la grabación, programas.



Núm. 7
Analizamos el Generador de Sonido. Aplicaciones matemáticas con el ordenador. La memoria de video. Trucos, noticias.



Núm. 8
Compact Disc, el periférico del futuro. Test: Dynadata DPC-200. Continuamos con la memoria de video. Libros, software, programas, trucos.



Núm. 9
Características técnicas del Compact Disc. Tratamiento de datos. Test: Quick Disk. Trucos, libros, noticias, programas.



PARA HACER SU PEDIDO, RELLENE ESTE CUPON, HOY MISMO Y ENVIELO A MSX MAGAZINE BRAVO MURILLO, 377. Tel. 733 96 62 - 28020 MADRID

Ruego me envíen los siguientes números atrasados ....

al precio de **250** ptas. cada uno. Cuyo importe abonaré:

☐ POR CHEQUE ☐ CONTRA REEMBOLSO ☐ CON MI TARJETA DE CREDITO

☐ AMERICAN EXPRESS ☐ VISA ☐ INTERBANK

Número de mi tarjeta ....

Fecha de caducidad ....

NOMBRE ....

DIRECCION ....

POBLACION .... C.P. ....

PROVINCIA ....

Código secreto

El que se sienta con ganas de romper cajas fuertes o el que tenga facilidad para abrir candados, le bastará con conectar el ordenador y ejecutar el programa.

En este juego hay que descubrir el código de cinco números capaz de abrir el candado que aparece en la patalla. Para ello tenemos que introducir el código separando los números mediante comas.

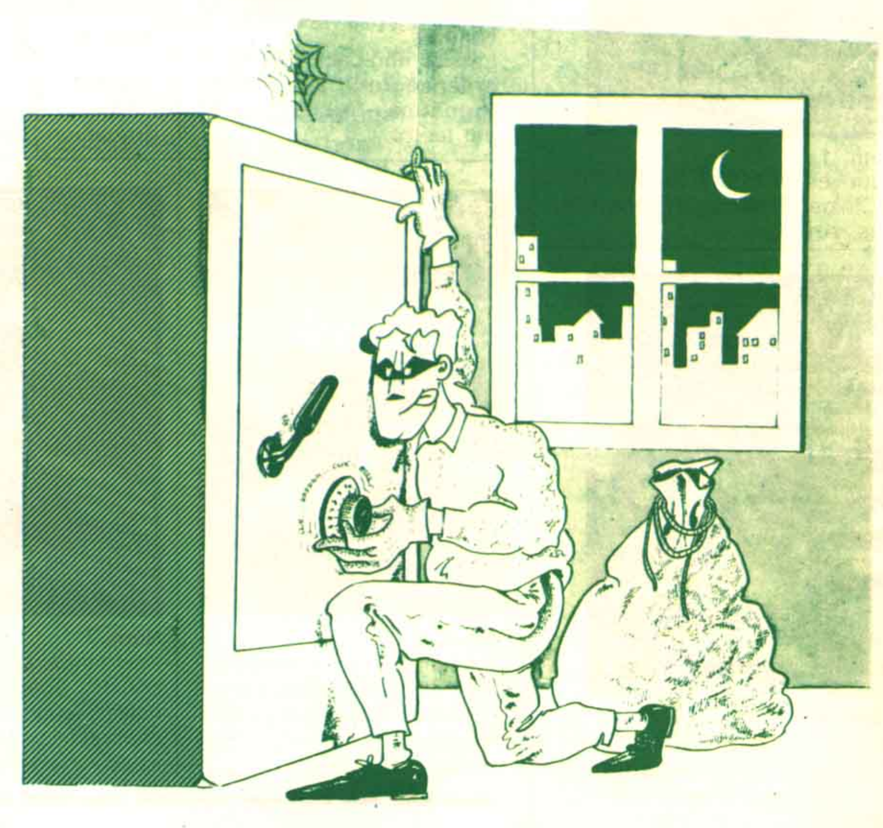
El candado se divide en tres partes: superior, inferior y central.

En la parte superior aparecerá un código que nosotros hayamos introducido, en la parte inferior aparecerán aquellos números que hayamos colocado en su lugar correcto y en la parte central veremos los números colocados incorrectamente.

Una vez conseguida la combinación correcta, el candado se abrirá y el ordenador nos preguntará si deseamos jugar de nuevo.

¡Ojo! no es tan fácil como parece.

José Gillén Fructuoso
Valencia

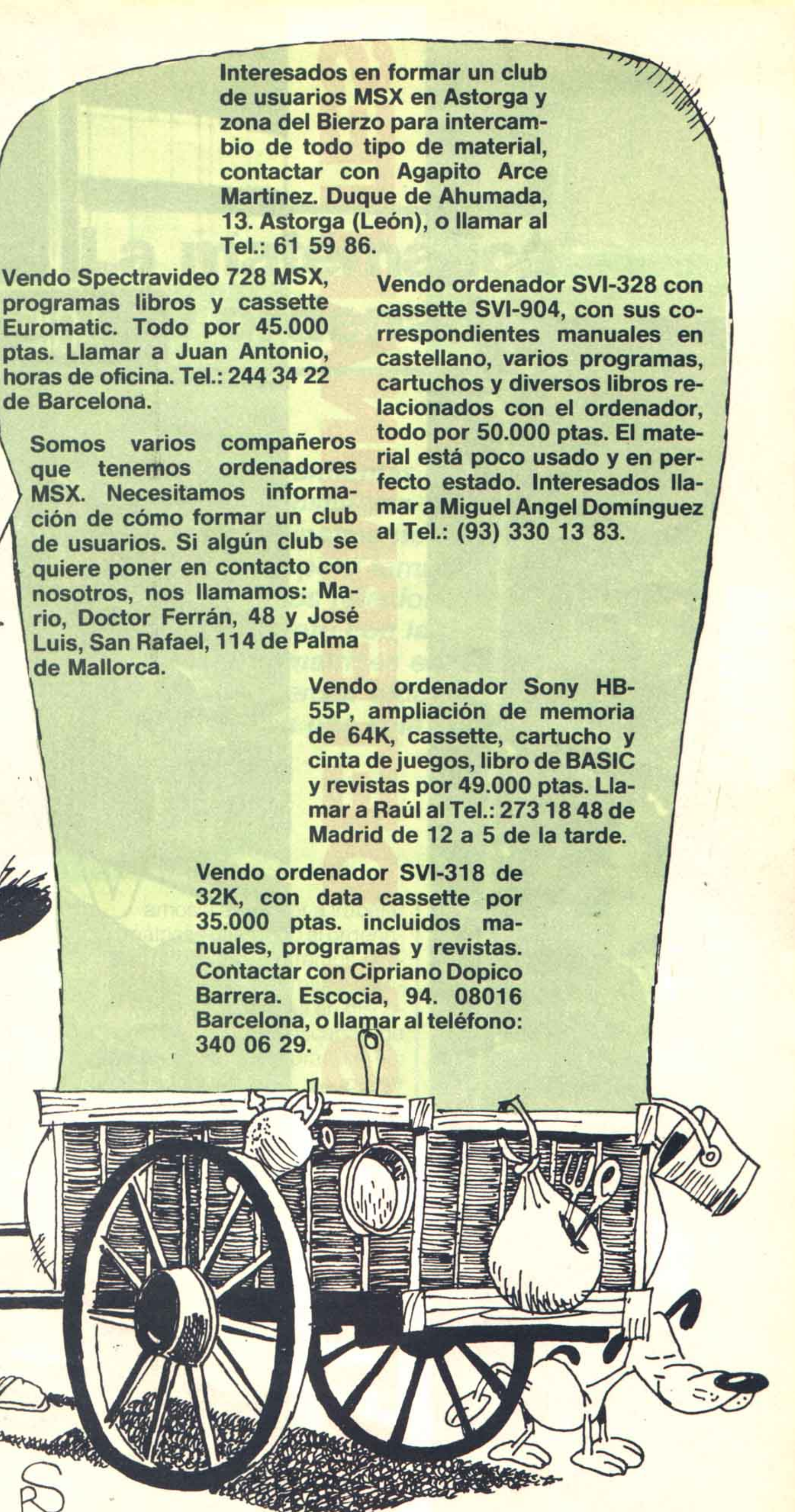


```
10 MAXFILES=2:CLS:SCREEN 2:COLOR 1,15,15:X=RND(-TIME)
20 FOR E=0 TO 2
30 G$="L8CBL16ABO5CO4BAGL8FGE"
40 PLAY G$
50 NEXT
60 F=INT(RND(1)*10)
70 G=INT(RND(1)*10)
80 H=INT(RND(1)*10)
90 I=INT(RND(1)*10)
100 J=INT(RND(1)*10)
110 F$=STR$(F):G$=STR$(G):H$=STR$(H)
120 I$=STR$(I):J$=STR$(J)
130 OPEN"GRP:"AS 1
140 PSET(20,20):PRINT#1,"INTENTOS QUE DESEA HACER"
150 FOR Q=0 TO 1000:NEXT
160 INPUT U
170 SCREEN 2:OPEN"GRP:"AS 2
180 FOR N=1 TO U
```

```
190 PSET(20,20):PRINT#2,"CLAVE DE CINCO NUMEROS.SEPARE    CON COMAS"
200 FOR I=0 TO 1000:NEXT
210 INPUT A,B,C,D,E
220 A$=STR$(A):B$=STR$(B):C$=STR$(C):D$=STR$(D):E$=STR$(E)
230 CLS:SCREEN 2
240 COLOR 11,15,15
260 CIRCLE(127,96),95,4,,,1.4
270 PAINT(127,96),4,4
280 CIRCLE(127,96),45,15,,,1.5
290 PAINT(127,96),15,15
300 LINE(58,97)-(195,191),1,BF
310 CIRCLE(76,177),5,4,,,1.4
320 PAINT(76,177),4,4
330 CIRCLE(76,115),5,4,,,1.4
340 PAINT(76,115),4,4
350 CIRCLE(177,115),5,4,,,1.4
360 PAINT(177,115),4,4
370 CIRCLE(177,177),5,4,,,1.4
380 PAINT(177,177),4,4
390 BEEP:BEEP
400 PSET(79,115):PRINT#1,A$+B$+C$+D$+E$
410 Z$=A$+B$+C$+D$+E$:X$=F$+G$+H$+I$+J$
420 IF Z$=X$ GOTO 760
430 Z$="   "
440 IF A=F THEN PSET(79,177):PRINT#1,A$+Z$+Z$+Z$+Z$
450 IF B=G THEN PSET(79,177):PRINT#1,Z$+B$+Z$+Z$+Z$
460 IF C=H THEN PSET(79,177):PRINT#1,Z$+Z$+C$+Z$+Z$
470 IF I$=D$ THEN PSET(79,177):PRINT#1,Z$+Z$+Z$+I$+Z$
480 IF E=J THEN PSET(79,177):PRINT#1,Z$+Z$+Z$+Z$+E$
490 Z$=" "
500 COLOR 4,15,15
510 IF A=G THEN PSET(79,140):PRINT#1,A$
520 IF A=H THEN PSET(79,140):PRINT#1,Z$+A$
530 IFA$=I$THEN PSET(79,140):PRINT#1,Z$+Z$+I$
540 IF A=J THEN PSET(79,140):PRINT#1,Z$+Z$+Z$+Z$+Z$+Z$+Z$+Z$+Z$+Z$+A$
550 IF B=F THEN PSET(79,140):PRINT#1,Z$+Z$+Z$+Z$+B$
560 IF B=H THEN PSET(79,140):PRINT#1,Z$+Z$+Z$+Z$+Z$+B$
570 IFB$=I$THEN PSET(79,140):PRINT#1,Z$+Z$+Z$+Z$+Z$+Z$+Z$+Z$+Z$+B$
580 IF B=J THEN PSET(79,150):PRINT#1,Z$+Z$+Z$+B$
590 IF C=F THEN PSET(79,150):PRINT#1,Z$+Z$+Z$+Z$+C$
600 IF C=G THEN PSET(79,150):PRINT#1,Z$+Z$+Z$+Z$+Z$+C$
610 IFC$=I$THEN PSET(79,150):PRINT#1,Z$+Z$+Z$+Z$+Z$+Z$+Z$+Z$+Z$+C$
620 IF C=J THEN PSET(79,150):PRINT#1,Z$+Z$+Z$+Z$+Z$+Z$+Z$+C$
630 IF D=F THEN PSET(79,160):PRINT#1,D$
640 IF D=G THEN PSET(79,160):PRINT#1,Z$+D$
650 IF D=H THEN PSET(79,160):PRINT#1,Z$+Z$+D$
660 IF D=J THEN PSET(79,160):PRINT#1,Z$+Z$+Z$+D$
670 IF E=F THEN PSET(79,160):PRINT#1,Z$+Z$+Z$+Z$+E$
680 IF E=G THEN PSET(79,160):PRINT#1,Z$+Z$+Z$+Z$+Z$+E$
690 IF E=H THEN PSET(79,160):PRINT#1,Z$+Z$+Z$+Z$+Z$+Z$+E$
700 IFE$=I$THEN PSET(79,160):PRINT#1,Z$+Z$+Z$+Z$+Z$+Z$+Z$+E$
710 FOR I=0 TO 10000:NEXT
730 COLOR 1,15,15
740 BEEP:BEEP:NEXT
750 GOTO 910
760 COLOR 1,15,15
770 PSET(20,20):PRINT#1,"ABIERTO!"
780 LINE(127,96)-(256,76),15,BF
790 CIRCLE(230,100),20,1,,,1.4
800 PAINT(230,100),1,1
810 CIRCLE(230,100),10,15,,,1.4
820 PAINT(230,100),15,15
```

```
830 LINE(228,115)-(232,170),1,BF
840 LINE(230,168)-(210,164),1,BF
850 LINE(230,158)-(215,154),1,BF
860 FOR E=0 TO 4
870 G$="L8CBL16ABO5CO4BAGL8FGE"
880 PLAY G$
890 NEXT
900 FOR I=0 TO 5000:NEXT:GOTO 1000
910 PSET(20,20):PRINT#1,"HA PERDIDO"
920 COLOR 11,15,15
930 PSET(79,177):PRINT#1,F$+G$+H$+I$+J$
940 FOR E=0 TO 5
950 F$="O3EL8DEFEDCO2L4BGO3C"
960 PLAY F$
970 NEXT
980 FOR I=0 TO 5000:NEXT
990 COLOR 1,15,15
1000 PSET(20,50):PRINT#1,"OTRA?":FOR I=0 TO 1000:NEXT
1010 K$=INKEY$:IF K$=""GOTO 1010
1020 IF K$="S" OR K$="s"GOTO 10
1030 SCREEN 0
1040 LOCATE 12,16:PRINT"OTRA VEZ SERA"
1050 FOR O=0 TO 1000:NEXT
1060 CLS:END
```

LOS JUEGOS ELECTRONICOS



Interesados en formar un club de usuarios MSX en Astorga y zona del Bierzo para intercambio de todo tipo de material, contactar con Agapito Arce Martínez. Duque de Ahumada, 13. Astorga (León), o llamar al Tel.: 61 59 86.

Vendo Spectravideo 728 MSX, programas libros y cassette Euromatic. Todo por 45.000 ptas. Llamar a Juan Antonio, horas de oficina. Tel.: 244 34 22 de Barcelona.

Somos varios compañeros que tenemos ordenadores MSX. Necesitamos información de cómo formar un club de usuarios. Si algún club se quiere poner en contacto con nosotros, nos llamamos: Mario, Doctor Ferrán, 48 y José Luis, San Rafael, 114 de Palma de Mallorca.

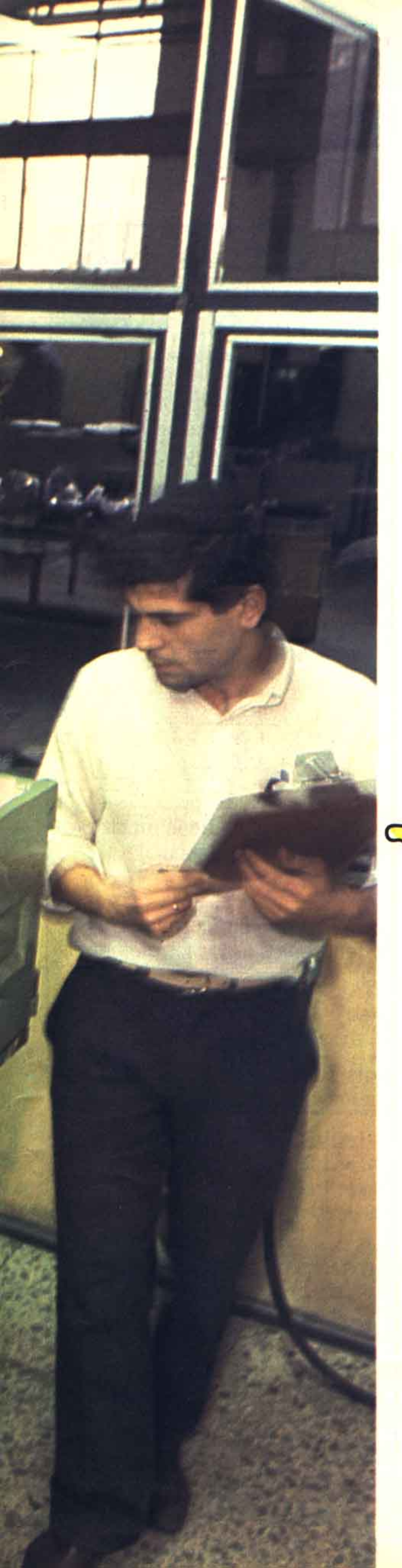
Vendo ordenador SVI-328 con cassette SVI-904, con sus correspondientes manuales en castellano, varios programas, cartuchos y diversos libros relacionados con el ordenador, todo por 50.000 ptas. El material está poco usado y en perfecto estado. Interesados llamar a Miguel Angel Domínguez al Tel.: (93) 330 13 83.

Vendo ordenador Sony HB-55P, ampliación de memoria de 64K, cassette, cartucho y cinta de juegos, libro de BASIC y revistas por 49.000 ptas. Llamar a Raúl al Tel.: 273 18 48 de Madrid de 12 a 5 de la tarde.

Vendo ordenador SVI-318 de 32K, con data cassette por 35.000 ptas. incluidos manuales, programas y revistas. Contactar con Cipriano Dopico Barrera. Escocia, 94. 08016 Barcelona, o llamar al teléfono: 340 06 29.

MATRICES Y DETERMINANTES





La matemática y el ordenador

Una de las herramientas más útiles en Análisis Numérico la constituyen las MATRICES. Estas son útiles principalmente para resolver sistemas de ecuaciones, pero también lo son para otros muchos aspectos relacionados fundamentalmente con la técnica. La principal utilidad de las matrices en este aspecto consiste en poder condensar toda una serie de relaciones matemáticas en una sola ecuación matricial.

Vamos a aprender a trabajar con matrices utilizando el ordenador. Para ello empezaremos definiendo lo que es una matriz, aunque no entraremos aquí en discutir conceptos que se escapan de su interés meramente práctico. No entraremos por tanto en definir una matriz como elemento de un espacio vectorial, cosa que por otra parte sería lo correcto, pero que probablemente os diga poco o nada sobre lo que queremos definir. Por consiguiente nos limitaremos a INTERPRETAR un lenguaje matemático y a preparar el camino para, más adelante, utilizarlo en la resolución de algunos problemas.

Tras definir lo que es una matriz, hablaremos de los símbolos y notación empleada para su tratamiento. Posteriormente sumaremos y multiplicaremos matrices, hallaremos la matriz traspuesta de una dada y su inversa, y por fin calcularemos el determinante de una matriz. Si estáis familiarizados con estos conceptos, podréis comprobar lo sencillo que resulta su tratamiento con el ordenador.

Qué es una matriz. Símbolos y subíndices

Podemos definir una matriz como un cuadro ordenado, o una ordenación bidimensional de números, dispuestos en FILAS y COLUMNAS.

La siguiente tabla es una matriz:

1	2	–5	4
–3	4	1	–3
2	–1	0	0

Tiene 3 filas (líneas horizontales) y 4 columnas (líneas verticales). Por tanto diremos que se trata de una matriz 3x4. Si la matriz tu-

viera **m** filas y **n** columnas diríamos que se trata de una matriz **m**x**n.**

Cuando hablamos de una matriz en general, conviene utilizar letras en lugar de números para representar cada uno de los elementos. Para representar un elemento de una matriz emplearemos letras minúsculas, a las que pondremos dos subíndices, el primero de ellos indicará la fila y el segundo la columna, es decir a_{ij}. Por ejemplo, a la matriz anterior, i variará entre 1 y 3 y j entre 1 y 4, de forma que:

i=1, j=1... $a_{ij}=1$
i=1, j=2... $a_{ij}=2$
.
.
.
i=2, j=1... $a_{ij}=-3$
.
.
i=3, j=4... $a_{ij}=0$

En una matriz de orden **m**x**n,** i varía de 1 a **m** y j lo hará de 1 a **n.**

Cuando hablemos de una matriz completa la representaremos con una letra mayúscula, A, o con una letra minúscula con sus subíndices, encerrada entre corchetes, de la forma: $[a_{ij}]$. De esta forma:

$$A=[a_{ij}]$$

Suma de matrices

La suma de dos matrices A y B es otra matriz C, tal que el elemento c_{ij} es igual al elemento a_{ij} más el elemento b_{ij}:

$$[c_{ij}]=[a_{ij}]+[b_{ij}]$$

Si el número de filas de la matriz A es **m** y el de columnas es **n,** la matriz B deberá tener el mismo número de filas y de columnas que A, y la matriz suma C tendrá el mismo número de filas y de columnas respectivamente que ambas.

Por tanto, la suma de las matrices:

$$A=\begin{matrix} 1 & 2 & 3 & 4 \\ -1 & 1 & 1 & 3 \\ 0 & -1 & 5 & 2 \end{matrix} \qquad B=\begin{matrix} 0 & -1 & 3 & 1 \\ 1 & 1 & 1 & 2 \\ 1 & -1 & 0 & 3 \end{matrix}$$

es la matriz:

$$C=\begin{matrix} 1 & 1 & 6 & 5 \\ 0 & 2 & 2 & 5 \\ 1 & -2 & 5 & 5 \end{matrix}$$

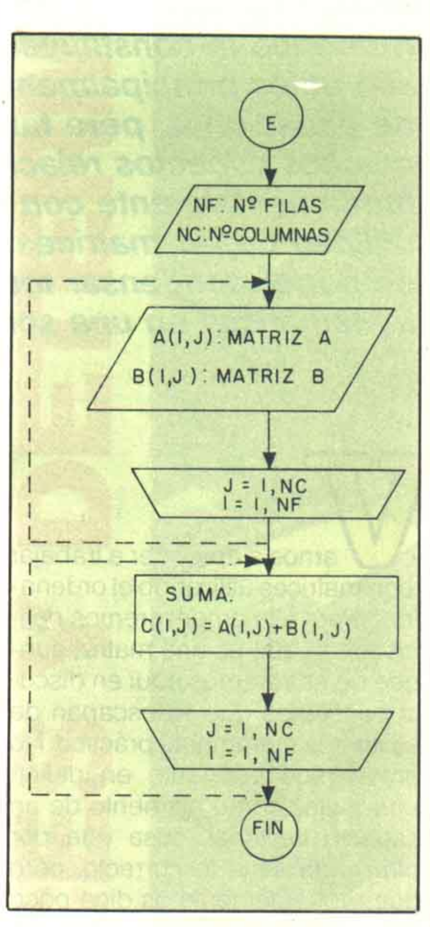


Diagrama 1: Suma de matrices.

donde:

1= 1+0 : 1= 2−1 : 6=3+3 : 5=4+1
0=−1+1 : 2= 1+1 : 2=1+1 : 5=3+2
1= 0+1 : −2=−1−1 : 5=5+0 : 5=2+3

Producto de matrices

El producto de dos matrices A y B se define como otra matriz C tal que un elemento c_{ij} de C es igual al producto uno a uno de los elementos de la fila i de la matriz A por los elementos de la columna j de la matriz B.

Por consiguiente, si la matriz A tiene **m** filas y la matriz B tiene **n** columnas, la matriz producto, C, tendrá **m** filas y **n** columnas. Además, para poder multiplicar todos los elementos de la fila i de la matriz A por todos los elementos de la columna j de la matriz B, el número de filas de la matriz B deberá ser igual al número de columnas de la matriz A.

Veamos esto con un ejemplo, que servirá para aclarar dudas. Sean:

$$A=\begin{matrix} 1 & 2 & 3 & 4 \\ -1 & 1 & 1 & 3 \\ 0 & -1 & 5 & 3 \end{matrix} \qquad B=\begin{matrix} 1 & 0 & -1 \\ 2 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & -1 & 2 \end{matrix}$$

El elemento c_{ij} de la matriz producto vendrá dado por:

$$c_{ij}=a_{ik}*b_{kj}$$

de forma que dando valores a i, j y k, tendremos todos los elementos de la matriz C:

$i=1, j=1 : c_{11} = a_{11} * b_{11} + a_{12} * b_{21} + a_{13} * b_{31} + a_{14} * b_{41}$

y así sucesivamente. Por tanto tendremos:

$c_{11} = 1*1 + 2*2 + 3*(-1) + 4*0 = 2$
$c_{12} = 1*0 + 2*0 + 3*1 + 4*(-1) = -1$
$c_{13} = 1*(-1) + 2*0 + 3*0 + 4*2 = 7$
$c_{21} = -1*1 + 1*2 + 1*(-1) + 3*0 = 0$
.
.
.
.
.
$c_{33} = 0*(-1) + (-1)*0 + 5*0 + 3*2 = 6$

Por tanto la matriz producto quedará:

$$C=\begin{matrix} 2 & -1 & 7 \\ 0 & -2 & 7 \\ -7 & 2 & 6 \end{matrix}$$

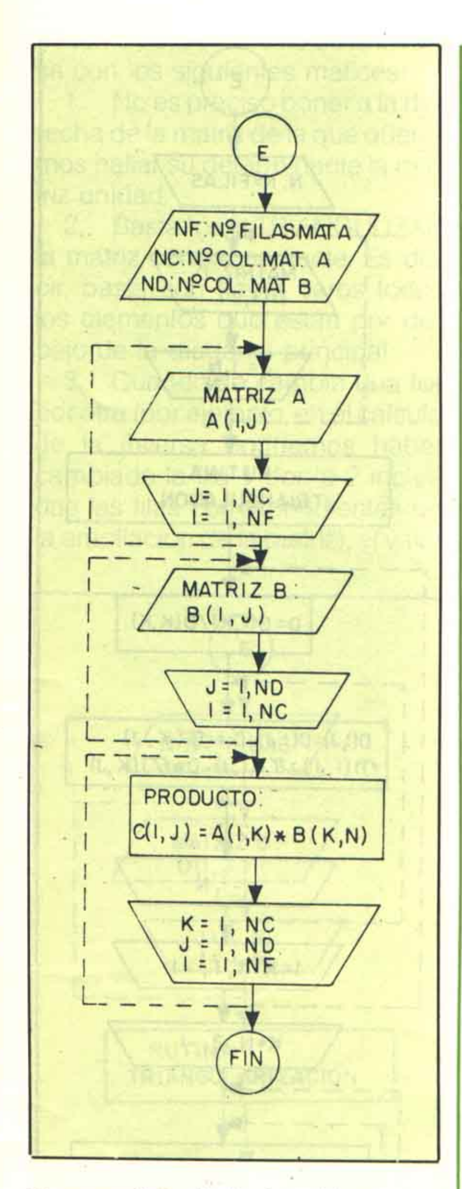


Diagrama 2: Producto de matrices.

Matriz traspuesta

La matriz traspuesta de una matriz dada:

$$A = [a_{ij}]$$

se obtiene simplemente cambiando las filas de A por sus columnas. Así si A tiene **m** filas y **n** columnas, su traspuesta, A^T tendrá **n** filas y **m** columnas:

$$A^T = [a_{ij}]$$

Por ejemplo, la traspuesta de la matriz A del apartado anterior, será:

$$AT= \begin{matrix} 1 & -1 & 0 \\ 2 & 1 & -1 \\ 3 & 1 & 5 \\ 4 & 3 & 3 \end{matrix}$$

Matriz inversa

La matriz inversa A^{-1} de una dada, A, es una matriz tal que:

$$A * A^{-1} = I$$

donde I es la matriz unidad:

$$I = [u_{ij}] : u_{ij} = 1 \text{ si } i = j$$
$$u_{ij} = 0 \text{ si } i <> j$$

Es decir, I es una matriz cuadrada (igual número de filas que de columnas) tal que todos los elementos de la diagonal son iguales a 1 y el resto de ellos son iguales a 0. Por ejemplo, la matriz unidad de 3x3 es:

$$I3= \begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix}$$

Hay varios métodos para calcular la inversa de una matriz. El más conocido consiste en expresar cada elemento de la inversa en función del ADJUNTO del elemento correspondiente de la matriz original dividido por el DETERMINANTE de dicha matriz. Sin embargo, este método no es muy útil a la hora de querer aplicarlo al ordenador (de hecho no es útil tampoco para aplicarlo manualmente).

El método que vamos a emplear es conocido como METODO DE GAUSS o METODO DE TRANSFORMACIONES ELEMENTALES. Explicaremos este método con un ejemplo:

Vamos a hallar la matriz inversa de la matriz C, resultado del ejemplo del apartado anterior. Para ello escribiremos la matriz C seguida de la matriz unidad, de la siguiente forma:

$$\begin{matrix} 2 & -1 & 7 & : & 1 & 0 & 0 \\ 0 & -2 & 7 & : & 0 & 1 & 0 \\ -7 & 2 & 6 & : & 0 & 0 & 1 \end{matrix}$$

Tenemos que convertir esto en algo de la forma:

$$\begin{matrix} 1 & 0 & 0 & : & a & b & c \\ 0 & 1 & 0 & : & d & e & f \\ 0 & 0 & 1 & : & g & h & I \end{matrix}$$

mediante transformaciones que consisten en sumar a cada fila una combinación lineal de las otras.

Como veis, se trata de convertir todos los elementos de la matriz en ceros excepto los de la diagonal principal, que deberán ser todos unos. Empezaremos por con-

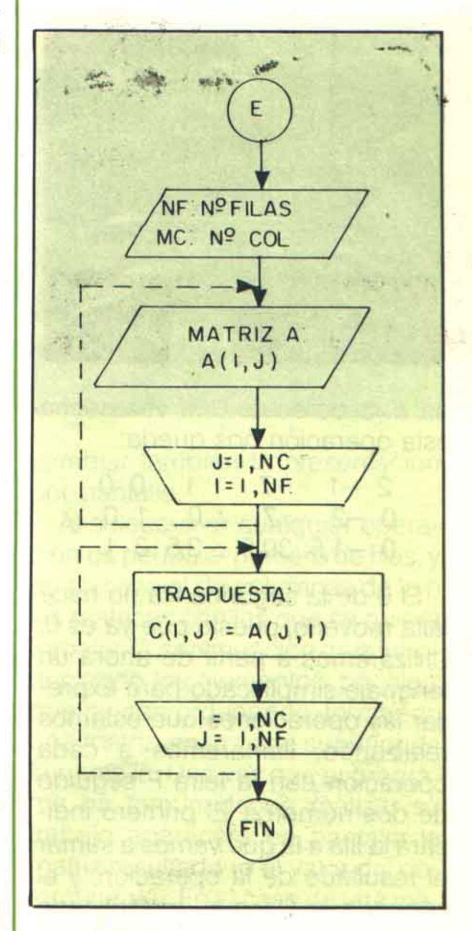
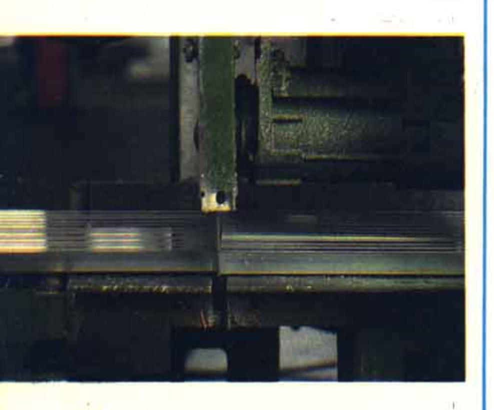


Diagrama 3: Matriz traspuesta.

vertir en cero el −7 de la tercera fila, primera columna. Para ello le sumamos a la fila 3, la fila 1 multiplicada por 7/2. Obsérvese que al dividir por 2 hemos cambiado el 2 de la fila 1, columna 1 por un 1, y al multiplicarlo por 7, lo hemos hecho por el −7 de la fila 3, columna 1, de forma que, al multiplicar 7/2 por 2 y sumarle −7 obtengamos un 0 en la posición que ahora ocupa el −7. Por tanto, resumiendo, multiplicamos los elementos de la fila 1 por 7/2 y se los sumamos a los correspondientes de la fila 3, sustituyendo éstos por el resultado. Esta operación debe realizarse, simultáneamente con los elementos de la matriz unidad, situa-



da a la derecha. Una vez hecha esta operación nos queda:

2	−1	7	:	1	0	0
0	−2	7	:	0	1	0
0	−1,5	30,5	:	3,5	0	1

El 0 de la segunda fila no hace falta moverlo, puesto que ya es 0. Utilizaremos a partir de ahora un lenguaje simplificado para expresar las operaciones que estamos realizando: llamaremos a cada operación con la letra F seguido de dos números. El primero indicará la fila a la que vamos a sumar el resultado de la operación, y el segundo la fila que, multiplicada por algo, vamos a sumar a la anterior. Ese «algo» lo escribiremos a continuación encerrado entre paréntesis. Así, la operación que ya hemos realizado (sumar a la tercera fila la primera multiplicada por 7/2) la escribiremos:

F31(7/2)

Ahora vamos a convertir en cero el −1,5 de la tercera fila, segunda columna, sumándole a los elementos de dicha fila, los de la fila 2 multiplicados por − 1,5/2, es decir:

F32(−1,5/2):

2	−1	7	:	1	0	0
0	−2	7	:	0	1	0
0	0	−25,25	:	3,5	−0,75	1

Seguidamente dividieremos todos los elementos de la fila 3 por − 25,25:

F3(−1/25,25):

2	−1	7	:	1	0	0
0	−2	7	:	0	1	0
0	0	1	:	−0,14	−0,03	−0.04

Ahora sumamos a la segunda fila la tercera multiplicada por −7:

F23(−7):

2	−1	7	:	1	0	0
0	−2	0	:	0,98	1,21	0
0	0	1	:	−0,14	− 0,03	−0,04

A continuación dividiremos la segunda fila por −2 (F2(−1/2)) y seguidamente sumaríamos a la primera la segunda multiplicada por 1 (F12(1)) y la tercera multiplicada por −7 (F13(−7)). Con esto habremos terminado de calcular la matriz inversa que buscamos y el resultado será:

1	0	0	:	0,257	−0,198	−0,693
0	1	0	:	0,485	−0,604	0,139
0	0	1	:	0,139	−0,030	0,040

con lo que a la derecha de la columna punteada tendremos la matriz inversa que nos interesa.

Determinantes

En cuanto a los determinantes, el proceso de cálculo es similar al

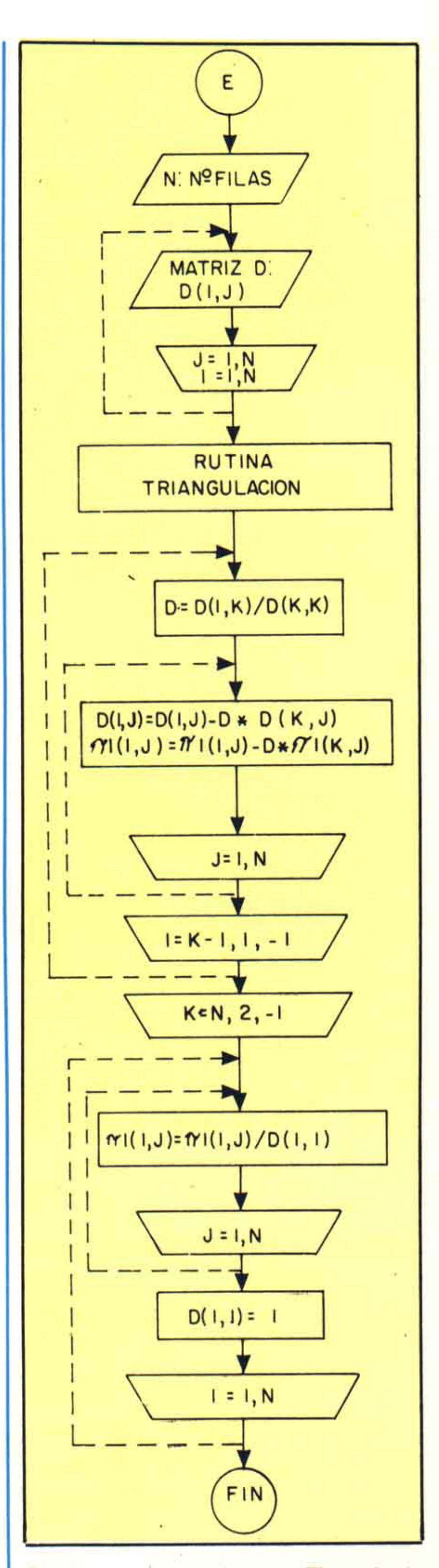


Diagrama 4: Matriz inversa. El resultado es la matriz MI(I,J).

descrito para el cálculo de la inversa con los siguientes matices:

1. No es preciso poner a la derecha de la matriz de la que queremos hallar su determinante la matriz unidad.

2. Basta con TRIANGULIZAR la matriz correspondiente. Es decir, basta con hacer ceros todos los elementos que están por debajo de la diagonal principal.

3. Cuando se cambie una fila por otra (por ejemplo, en el cálculo de la inversa podríamos haber cambiado la fila 1 por la 2 incluidas las filas correspondientes de la ampliación de la matriz), el valor

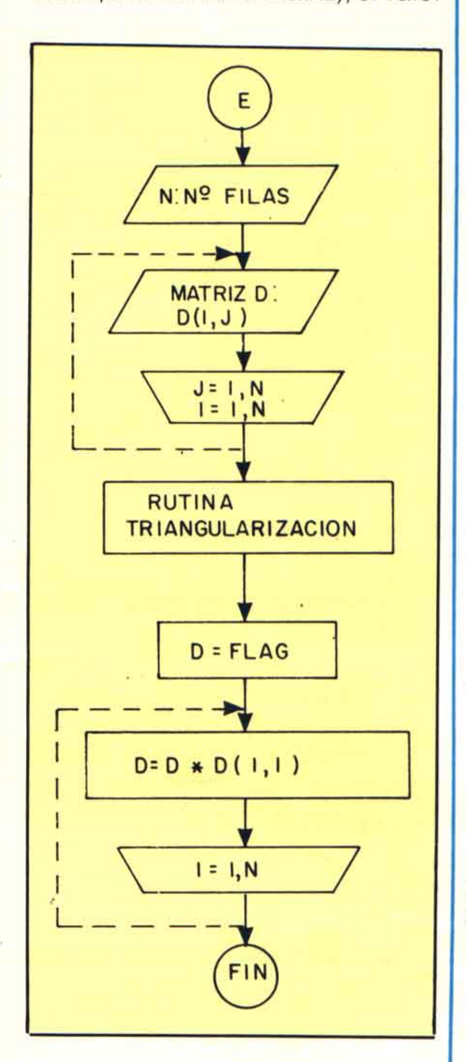
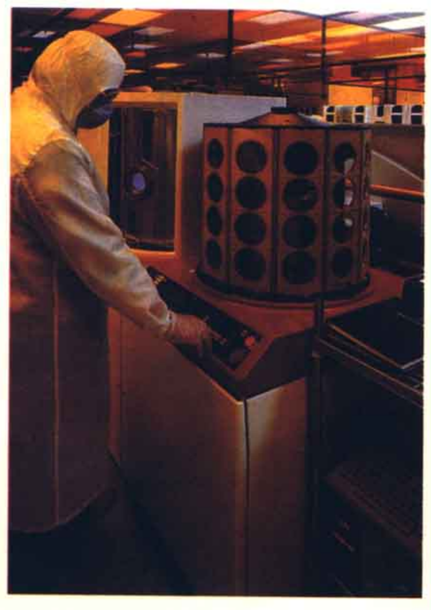


Diagrama 5: Determinante.



del determinante cambia de signo.

Por consiguiente, para calcular el determinante de la matriz C del ejemplo anterior, habríamos realizado los pasos 1 y 2 y habríamos llegado a:

2	−1	7
0	−2	7
0	0	−25,25

En este punto, basta multiplicar entre sí los elementos de la diagonal principal, teniendo en cuenta los cambios de signo (cambios de filas) realizados. En el ejemplo, como no hemos cambiado de fila, no tenemos que cambiar el signo, y el valor del determinante será:

$$D = 2 \cdot (-2) \cdot (-25{,}25) = 101$$

El que se pueda calcular un determinante de esta forma tan sencilla (olvidando por tanto el tener que desarrollar por una fila, multiplicando por los correspondientes adjuntos, etc.) nos permitirá calcular con rapidez determinantes de 6 filas por 6 columnas o más, y esto se puede hacer así por simple aplicación de dos reglas de los determinantes:

1. Si a una fila de un determinante le sumamos una combinación lineal de las restantes filas, el determinante no varía.

2. Si en un determinante se intercambian dos filas (o columnas) el determinante cambia de signo.

El programa

El programa que se lista a continuación puede realizar cualquiera de las operaciones que hemos visto entre matrices, y el cálculo de la inversa y del determinante de una matriz cuadrada. Solamente tiene una limitación y es que admite un número máximo de 6 filas y 6 columnas. Evidentemente podéis modificarlo para trabajar con un número mayor de filas y columnas, pero entonces tendréis que



cambiar también la presentación por pantalla.

Al seleccionar cualquier operación os pedirá el número de filas, y en su caso el de columnas de la o las matrices con las que se quiera trabajar. A continuación iréis introduciendo los elementos, por filas, que se irán ordenando, formando una matriz, en la parte superior de la pantalla. Una vez que el programa ha terminado de realizar su trabajo aparecerá en pantalla la matriz resultado (o el valor del determinante). En el caso de una matriz, los elementos resultados aparecerán recortados, teniendo un solo decimal. Podéis listar todos

los elementos pulsando la tecla F5, y después cualquier tecla para obtener los elementos uno a uno. Una vez listados todos, se puede volver al estado anterior pulsando RETURN. Pulsando ahora cualquier tecla el programa volverá al MENU.

Si estáis calculando una matriz inversa o un determinante, y la matriz es SINGULAR, es decir, su determinante es cero, os aparecerá un mensaje de matriz singular, y, pulsando cualquier tecla, volveréis al MENU.

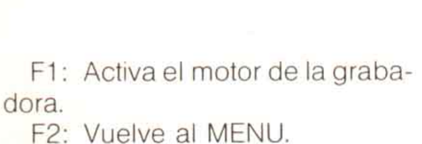
Finalmente, tenéis cinco teclas de control para lo siguiente:

```
10 'MATRICES Y DETERMINANTES
20 '=======================
30 '
40 'Versión 15.12.85- 4499 bytes
50 'Juan Antonio Feberero Castejón
60 '
70 SCREEN 0..0:KEY OFF:WIDTH 39
80 CLEAR 200
90 DEFINT I-K.N
100 ON KEY GOSUB 1580.1590.1600.1610.1650
110 KEY(1) ON:KEY(2) ON:KEY(3) ON
120 ON STOP GOSUB 1600
130 STOP ON
140 ON ERROR GOTO 1630
150 '
160 'MENU
170 '====
180 '
190 CLS:RESTORE 1660:READ A$:LOCATE 7.5:PRINT A$.STRING$(39.195)
200 DIM A(6.6).B(6.6).C(6.6).D(6.6).MI(6.6):FLAG=0
210 FOR I=1 TO 5:READ A$:LOCATE 7.6+2*I:PRINT USING "<#> &":I:A$:NEXT
220 GOSUB 1370
230 W$=INKEY$:IF W$="" THEN 230
240 IF INSTR("12345".W$)=0 THEN 230
250 INDICE=VAL(W$):GOSUB 1430
260 '
270 'ENTRADA DATOS
280 '=============
290 '
300 PRINT "Se admite un número máximo de 6          filas y 6 columnas."...
310 IF INDICE=2 THEN 350
320 INPUT "Número de filas":NF:IF NF>6 THEN GOSUB 1480:GOTO 320
330 IF INDICE>3 THEN NC=NF:GOTO 370
340 INPUT "Número de columnas":NC:IF NC>6 THEN GOSUB 1480:GOTO 340 ELSE 400
350 INPUT "Número de filas matriz <A>":NF:IF NF>6 THEN GOSUB 1480:GOTO 350
360 INPUT "Número de columnas matriz <A>";NA:IF NA>6 THEN GOSUB 1480:GOTO 360
370 ON INDICE GOTO 380.380.400.400.400'ENTRADAS
380 INPUT "Número de columnas matriz <B>":NC:IF NC>6 THEN GOSUB 1480:GOTO 380
390 IF INDICE=2 THEN SWAP NA.NC
400 A$="A":GOSUB 930
410 FOR I=1 TO NF:FOR J=1 TO NC:A(I.J)=AUX(I.J):NEXT:NEXT
420 ON INDICE GOTO 440.430.640.700.820'SUBPRGM
430 SWAP NA.NC:SWAP NF.NA
440 A$="B":GOSUB 930
450 FOR I=1 TO NF:FOR J=1 TO NC:B(I.J)=AUX(I.J):NEXT:NEXT
460 IF INDICE=2 THEN 570
470 '
480 'SUMA
490 '====
500 '
510 FOR I=1 TO NF:FOR J=1 TO NC:C(I.J)=A(I.J)+B(I.J):NEXT:NEXT
520 GOTO 1260
```

```
530 '
540 'PRODUCTO
550 '========
560 '
570 SWAP NA,NF
580 FOR I=1 TO NF:FOR J=1 TO NC:C(I,J)=0:FOR K=1 TO NA:C(I,J)=C(I,J)+A(I,K)*B(K,
J):NEXT:NEXT:NEXT
590 GOTO 1260
600 '
610 'TRASPUESTA
620 '==========
630 '
640 FOR I=1 TO NF:FOR J=1 TO NC:C(J,I)=A(I,J):NEXT:NEXT
650 SWAP NF,NC:GOTO 1260
660 '
670 'INVERSA
680 '=======
690 '
700 FOR I=1 TO NF:FOR J=1 TO NF:SWAP D(I,J),A(I,J):NEXT:MI(I,I)=1:NEXT
710 GOSUB 1080
720 FOR K=NF TO 2 STEP -1:FOR I=K-1 TO 1 STEP -1
730 D=D(I,K)/D(K,K)
740 FOR J=NF TO 1 STEP -1:D(I,J)=D(I,J)-D*D(K,J):MI(I,J)=MI(I,J)-D*MI(K,J)
750 NEXT J:NEXT I:NEXT K
760 FOR I=1 TO NF:FOR J=1 TO NF:C(I,J)=MI(I,J)/D(I,I):NEXT:D(I,I)=1:NEXT
770 GOTO 1260
780 '
790 'DETEMINANTE
800 '===========
810 '
820 FOR I=1 TO NF:FOR J=1 TO NF:SWAP D(I,J),A(I,J):NEXT:NEXT
830 GOSUB 1080
840 DET=FLAG
850 FOR I=1 TO NF:DET=DET*D(I,I):NEXT
860 LOCATE 0,19:GOSUB 1480:LOCATE 0,20:GOSUB 1480
870 PRINT "DETERMINANTE=";DET
880 GOTO 1320
890 '
900 'Intro datos
910 '-----------
920 '
930 GOSUB 1430:GOSUB 1530
940 LOCATE 0,17:PRINT USING "MATRIZ <&>";A$
950 FOR I=1 TO NF:FOR J=1 TO NC
960 LOCATE 0,20:GOSUB 1480:PRINT USING "Término #;#";I;J;:INPUT A$:AUX(I,J)=VAL(
A$)
970 IF AUX(I,J)=0 AND A$<>"0" THEN 960
980 IF AUX(I,J)>999 OR AUX(I,J)<-99 THEN LOCATE 0,20:GOSUB 1480:PRINT "¡¡VALOR D
EMASIADO ALTO!!":FOR IAUX=1 TO 50:BEEP:NEXT:GOTO 960
990 LOCATE 1+6*(J-1),2*(I-1):PRINT USING "###.#";AUX(I,J)
1000 IF FLAG=1 THEN FLAG=0:RETURN 1020
1010 NEXT J:NEXT I
1020 LOCATE 0,20:GOSUB 1480:KEY(4) ON:PRINT "COMPRUEBA LOS DATOS....Pulsa una te
cla"
1030 IF INKEY$="" THEN 1030 ELSE RETURN
1040 '
1050 'Trianqulización
1060 '---------------
1070 '
1080 FLAG=1:FOR K=1 TO NF-1:FOR I=K+1 TO NF
1090 IF D(K,K)=0 THEN KK=K:GOSUB 1200
1100 D=D(I,K)/D(K,K)
1110 FOR J=1 TO NF:D(I,J)=D(I,J)-D*D(K,J):IF INDICE=4 THEN MI(I,J)=MI(I,J)-D*MI(
K,J)
```

```
1120 NEXT J
1130 NEXT I
1140 NEXT K
1150 RETURN
1160 '
1170 'Cambio de fila
1180 '--------------
1190 '
1200 KK=KK+1:FLAG=-FLAG:IF D(KK,K)=0 AND KK<NF THEN 1200 ELSE IF D(KK,K)=0 AND K
K=NF THEN RETURN 1140
1210 FOR J=1 TO NF:SWAP D(K,J),D(KK,J):SWAP MI(K,J),MI(KK,J):NEXT:RETURN
1220 '
1230 'Matriz resultado
1240 '----------------
1250 '
1260 GOSUB 1430:GOSUB 1530:LOCATE 0,17:PRINT "MATRIZ RESULTADO
1270 KEY(5) ON
1280 FOR I=1 TO NF:FOR J=1 TO NC
1290 LOCATE 1+6*(J-1),2*(I-1):PRINT USING "###.#";C(I,J)
1300 NEXT J:NEXT I
1310 LOCATE 0,19:PRINT "Pulsa una tecla para volver a MENU
1320 IF INKEY$="" THEN 1320 ELSE 80
1330 '
1340 'Leyenda
1350 '-------
1360 '
1370 LOCATE 0,21:PRINT STRING$(39,195);
1380 LOCATE 0,22:PRINT "F1:Motor    |F2:Menú      |F3:Fin","F4:Modificar|F5:Lista
r Result.";:LOCATE 0,0:RETURN
1390 '
1400 'Borra zona superior
1410 '-------------------
1420 '
1430 FOR I=1 TO 840:VPOKE I,0:NEXT:LOCATE 0,0:RETURN
1440 '
1450 'Borra línea
1460 '-----------
1470 '
1480 LOCATE 0,CSRLIN-1:PRINT STRING$(39,32);:LOCATE 0,CSRLIN-1:RETURN
1490 '
1500 'Barras matriz
1510 '-------------
1520 '
1530 NTB=NC*6:FOR I=1 TO 2*NF-1:LOCATE 0,I-1:PRINT "▌";TAB(NTB);"▌":NEXT:RETURN
1540 '
1550 'Desvíos
1560 '-------
1570 '
1580 MOTOR:RETURN
1590 RETURN 80
1600 CLS:STOP:RETURN 80
1610 FLAG=1:LOCATE 0,20:GOSUB 1480:LINE INPUT "FILA , COLUMNA? ";A$:J=INSTR(A$,"
,"):IF J=0 THEN 1610 ELSE I=VAL(LEFT$(A$,J)):J=VAL(MID$(A$,J+1)):IF I<1 OR J<1 O
R I>NF OR J>NC THEN 1610
1620 LOCATE 1+6*(J-1),2*(I-1):PRINT "     ":GOTO 960
1630 CLS:LOCATE 8,11:PRINT "ERROR:";:IF ERR=11 THEN PRINT " MATRIZ SINGULAR
1640 IF INKEY$="" THEN BEEP:GOTO 1640 ELSE 80
1650 KEY(5) ON:LOCATE 0,20:FOR I=1 TO NF:FOR J=1 TO NC
1660 GOSUB 1480:PRINT USING "Elemento #;#: &";I;J;STR$(C(I,J))
1670 IF INKEY$="" THEN 1670
1680 NEXT:NEXT
1690 IF INKEY$<>CHR$(13) THEN 1690 ELSE RETURN 1310
1700 DATA MATRICES Y DETERMINANTES,SUMA DE MATRICES,PRODUCTO DE MATRICES,MATRIZ
TRASPUESTA,MATRIZ INVERSA,CALCULO DE DETERMINANTES
```





F1: Activa el motor de la grabadora.

F2: Vuelve al MENU.

F3: Interrumpe el programa. Podéis continuar tecleando CONT.

F4: Permite una vez introducidos los datos, corregirlos si hubiese algún error.

F5: Permite listar los elementos de la matriz resultado uno a uno.

A continuación tenéis los diagramas de flujo de cada uno de los apartados anteriores, que no requieren más comentarios.

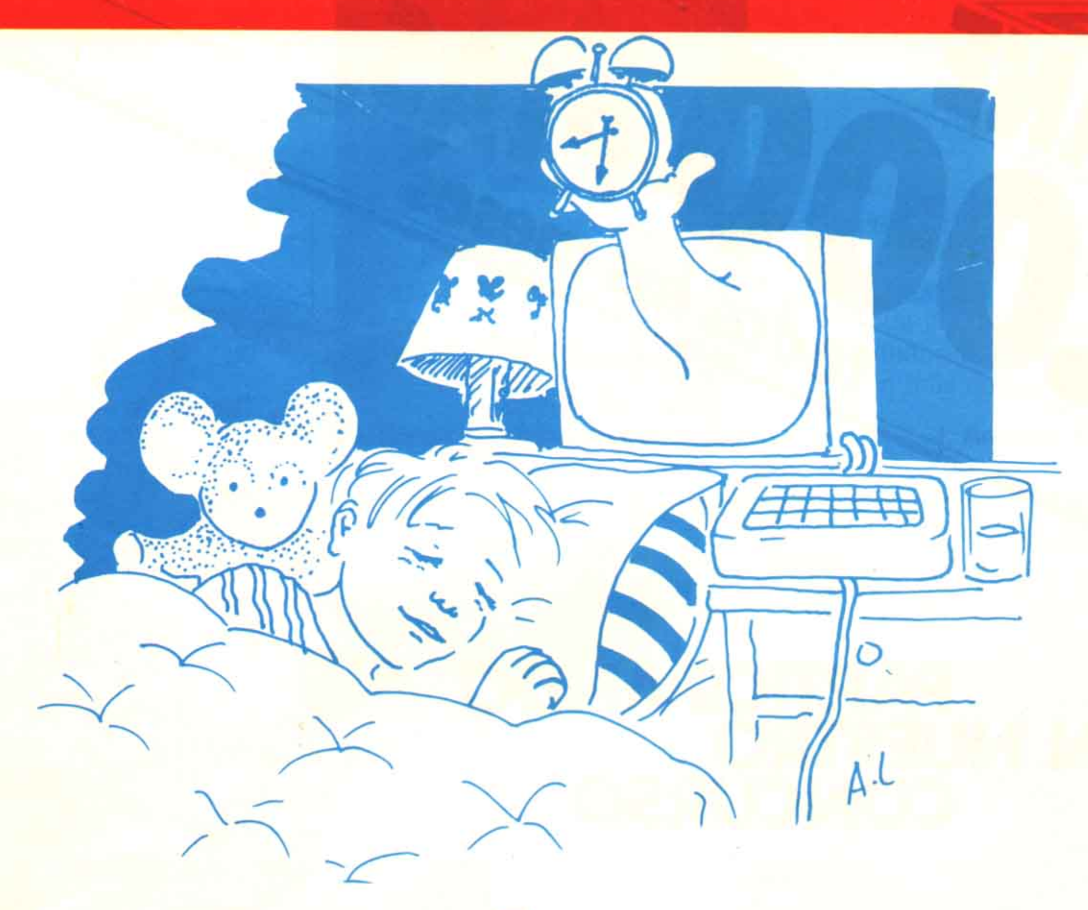
J. Antonio Feberero

DEL PROGRAMA PRINCIPAL
D(K,K)=O
SI
KK=K
NO
D=D(I,K)/D(K,K)
KK=KK+1
D(KK,KK)=O
KK=N
D(I,J)=D(I,J)-D* D(K,J)
NO
FLAG=-FLAG
SOLO EN MATRIZ INVERSA
MI(I,J)=MI(I,N)-D*MI(K,J)
SWAP D(K,J), D(KK,J)
SOLO EN MATRIZ INVERSA
SWAP MI(K,J), MI(KK,J)
J=1,N
I=K+1, N
J=1,N
K=1,N-1
AL PROGRAMA PRINCIPAL

Diagrama 6: Rutina de triangularización.



GANE
7.000 PTAS.
todos los meses
PARTICIPANDO
EN NUESTRO
CONCURSO
A partir del próximo número MSX premiará mensualmente los programas que hagan llegar los lectores.
Para participar en este concurso abierto, todo aficionado a los ordenadores con este estándar deberá hacer llegar a la redacción de la revista el listado, un cassette y un texto explicativo.
Entre todos los programas que recibamos cada mes, serán seleccionados para su publicación aquellos que reúnan los siguientes criterios:
— Originalidad de la aplicación.
— Simplicidad del método de programación.
La única condición para participar en el concurso será que los programas no hayan sido publicados previamente en ninguna revista.



Despertador musical

Una buena forma de comenzar el día... con música. Al ejecutar el programa, éste le pedirá la hora para el despertador, la hora para el reloj y la melodía que desea escuchar al despertar (hay cinco melodías). Para cambiar la melodía utilice las teclas de cursor «izquierda» y «derecha». Una vez elegida, pulse [RETURN].

Ahora el reloj está listo para funcionar. Para sincronizarlo con su reloj de pulsera o con la radio, pulse la tecla [RETURN] y el reloj se pondrá en marcha... y a dormir.

Las teclas de función están programadas del siguiente modo:

[F1]: Cambio de la hora de despertar.
[F2]: Cambio de la hora actual.
[F3]: Elección de melodía.
[F4]: Activación/Desactivación de la pantalla.
[F5]: Reinicialización del reloj.

La barra espaciadora se utiliza para que deje de sonar el despertador.

El programa se divide en los siguientes bloques:

Líneas 10 - 190: Inicialización de los sprites y variables.

Líneas 200 - 250: Preparación del reloj por el usuario.

Líneas 260 - 270: Activación.

Líneas 280 - 300: Chequeo de la hora y del despertador.

Líneas 310 - 330: Actualización de las horas y minutos.

Líneas 340 - 360: Subrutina "INPUT"

Líneas 370 - 450: Subrutina "Hora de despertar?"

Líneas 460 - 570: Subrutina "Hora actual?"

Líneas 580 - 640: Subrutina "elija una melodía:"

Línea 900: Desactiva la música.

Líneas 1000 - 1010: Ejecuta la melodía contenida en los DATAS.

Líneas 1100 - 1270: Activa/Desactiva pantalla.

Líneas 2000 - 2040: Bloque de RESTORES. se elige así una melodía según el valor de la variable "MEL".

Líneas 3000 - 3120: DATAS de "Singing in the Rain".

Líneas 4000 - 4080: DATAS de "Minue".

Líneas 5000 - 5080: DATAS de "Nana".

Líneas 6000 - 6240: DATAS de "Noche de Paz".

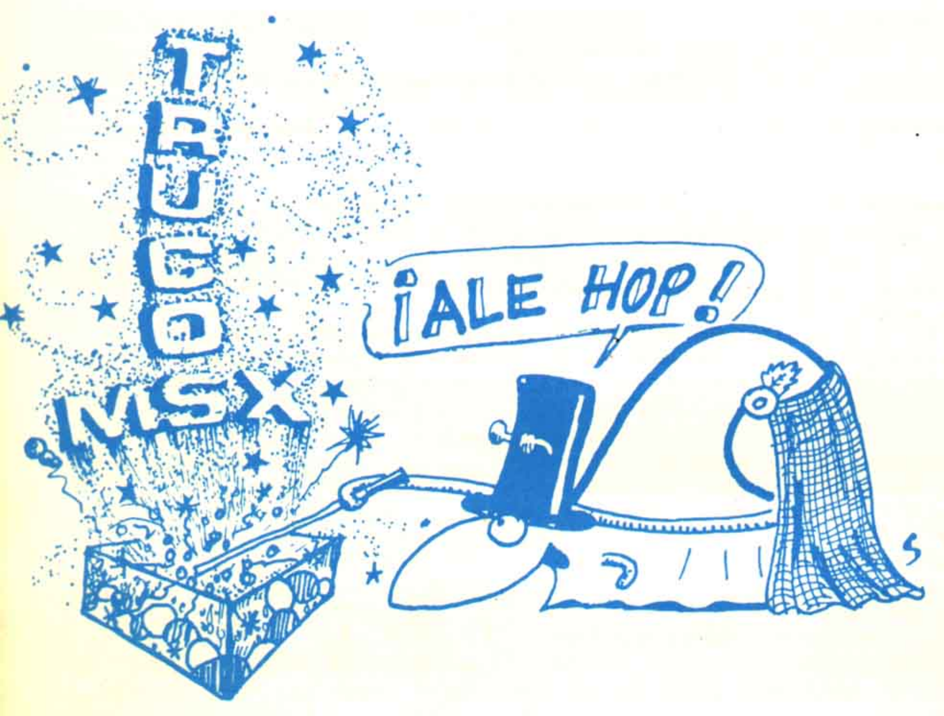
Líneas 7000 - 7080: DATAS de "Chotis".

Angel Zarazaga
Madrid

```
10 KEY OFF:DEFUSR=&H90:DEFUSR1=&H156
20 SCREEN 1,3:RESTORE
30 LOCATE 3,12:PRINT "UN MOMENTO, POR FAVOR"
40 FOR SP%=0 TO 10:FOR LAZO=1 TO 32
50 READ D$:D%=VAL("&h"+D$)
60 SP$=SP$+CHR$(D%):NEXT LAZO
70 SPRITE$(SP%)=SP$:SP$="":NEXT SP%
80 DATA 3f,3f,f0,f0,f0,f0,f3,f3,fc,fc,f0,f0,3f,3f,00,00,f0,f0,3c,3c,fc,fc,3c,3c,
3c,3c,3c,3c,f0,f0,00,00:REM 0
90 DATA 03,03,0f,0f,03,03,03,03,03,03,03,03,3f,3f,00,00,c0,c0,c0,c0,c0,c0,c0,c0,
c0,c0,c0,c0,fc,fc,00,00:REM 1
100 DATA 0f,0f,3c,3c,00,00,0f,0f,3c,3c,3c,3c,3f,3f,00,00,f0,f0,3c,3c,3c,3c,f0,f0
,00,00,3c,3c,fc,fc,00,00:REM 2
110 DATA 0f,0f,30,30,00,00,03,03,00,00,3c,3c,0f,0f,00,00,f0,f0,3c,3c,3c,3c,f0,f0
,3c,3c,3c,3c,f0,f0,00,00:REM 3
120 DATA 03,03,0f,0f,33,33,c3,c3,ff,ff,03,03,0f,0f,00,00,c0,c0,c0,c0,c0,c0,c0,c0
,fc,fc,c0,c0,f0,f0,00,00:REM 4
130 DATA 3f,3f,3c,3c,3c,3c,0f,0f,00,00,3c,3c,0f,0f,00,00,fc,fc,0c,0c,00,00,f0,f0
,3c,3c,3c,3c,f0,f0,00,00:REM 5
140 DATA 0f,0f,3c,3c,3c,3c,3f,3f,3c,3c,3c,3c,0f,0f,00,00,f0,f0,3c,3c,00,00,f0,f0
,3c,3c,3c,3c,f0,f0,00,00:REM 6
150 DATA 3f,3f,30,30,00,00,00,00,03,03,03,03,03,03,00,00,fc,fc,3c,3c,3c,3c,f0,f0
,c0,c0,c0,c0,c0,c0,00,00:REM 7
160 DATA 0f,0f,3c,3c,3c,3c,0f,0f,3c,3c,3c,3c,0f,0f,00,00,f0,f0,3c,3c,3c,3c,f0,f0
,3c,3c,3c,3c,f0,f0,00,00:REM 8
170 DATA 0f,0f,3c,3c,3c,3c,0f,0f,00,00,3c,3c,0f,0f,00,00,f0,f0,3c,3c,3c,3c,fc,fc
,3c,3c,3c,3c,f0,f0,00,00:REM 9
180 DATA 00,00,00,00,03,03,03,03,00,00,03,03,03,03,00,00,00,00,00,00,c0,c0,c0,c0
,00,00,c0,c0,c0,c0,00,00:REM :
190 FOR N=0 TO 4:READ NOM$(N):NEXT
200 DATA Singing in the Rain,Minue,Nana,Noche de Paz,Chotis
210 CLS:GOSUB 370
220 GOSUB 460
230 GOSUB 580
240 ON MEL+1 GOSUB 2000,2010,2020,2030,2040
250 LOCATE 0,21:PRINT"Pulse [RETURN] para empezar:"
260 K$=INKEY$:IF K$<>CHR$(13) THEN 260 ELSE ON INTERVAL=3000 GOSUB 310:INTERVAL
ON:LOCATE 0,21:PRINT SPACE$(28):
270 ON KEY GOSUB 370,460,580,1100,10:KEY(1) ON:KEY(2) ON:KEY(3) ON:KEY(4) ON:KEY
(5) ON:ON STRIG GOSUB 900:STRIG(0) ON
280 IF SUENA=1 THEN GOSUB 1000
290 IF M2=D2 AND M1=D1 AND H2=E2 AND H1=E1 AND DES=1 AND SUENA=0 THEN ON MEL+1 G
OSUB 2000,2010,2020,2030,2040:SUENA=1
300 GOTO 280
310 M2=(M2+1) MOD 10:PUT SPRITE 0,(180,70),15,M2:IF M2<>0 THEN 330 ELSE M1=(M1+1
) MOD 6:PUT SPRITE 1,(146,70),15,M1:IF M1<>0 THEN 330
320 H2=-((H1<2)*(H2+1)MOD 10)-((H1=2)*(H2+1)MOD 4):PUT SPRITE 2,(72,70),15,H2:IF
 H2<>0 THEN 330 ELSE H1=(H1+1)MOD 3:PUT SPRITE 3,(38,70),15,H1
330 RETURN
340 A=USR1(0):LOCATE 0,22:X=1:D$=SPACE$(L)
350 K$=INKEY$:IF K$="" THEN 350 ELSE IF K$=CHR$(8) AND X>1 THEN PRINT CHR$(127);
:X=X-1:MID$(D$,X,1)=" ":GOTO 350 ELSE IF K$=CHR$(13) THEN LOCATE 0,22:PRINT SPAC
E$(L):RETURN ELSE IF K$=CHR$(8) THEN 350 ELSE MID$(D$,X,1)=K$:PRINT K$;
360 IF X=L THEN LOCATE 0,22:PRINT SPACE$(L):RETURN ELSE X=X+1:GOTO 350
370 LOCATE 0,21:PRINT"Hora de despertar(HH/MM)?"
380 L=5:GOSUB 340:DH$=D$
390 LOCATE 0,21:PRINT SPACE$(50)
400 IF INSTR(DH$," ") THEN DES=0:DH$="00/00" ELSE DES=1
410 D2=VAL(MID$(DH$,5,1))
420 D1=VAL(MID$(DH$,4,1))
430 E2=VAL(MID$(DH$,2,1))
```

```
440 E1=VAL(MID$(DH$,1,1))
450 RETURN
460 LOCATE 0,21:PRINT"Hora actual(HH/MM)?"
470 L=5:GOSUB 340:AH$=D$
480 IF INSTR(AH$," ") THEN A=USR1(0):LOCATE 0,22:PRINT SPACE$(L):GOTO 470 ELSE L
OCATE 0,21:PRINT SPACE$(50)
490 M2=VAL(MID$(AH$,5,1))
500 M1=VAL(MID$(AH$,4,1))
510 H2=VAL(MID$(AH$,2,1))
520 H1=VAL(MID$(AH$,1,1))
530 PUT SPRITE 0,(180,70),15,M2
540 PUT SPRITE 1,(146,70),15,M1
550 PUT SPRITE 2,(72,70),15,H2
560 PUT SPRITE 3,(38,70),15,H1
561 LOCATE 13,9:PRINT CHR$(220)
562 LOCATE 13,10:PRINT CHR$(223)
563 LOCATE 13,11:PRINT CHR$(219)
570 RETURN
580 LOCATE 0,21:PRINT"Por favor, elija una melodia:":A=MEL
590 LOCATE 0,22:PRINT USING"##&&";A+1;". ";NOM$(A)
600 K$=INKEY$:IF K$="" THEN 600
610 IF K$=CHR$(28) THEN A=(A+1) MOD 5
620 IF K$=CHR$(29) THEN IF A>0 THEN A=A-1:GOTO 640 ELSE A=4:GOTO 640
630 IF K$=CHR$(13) THEN MEL=A:LOCATE 0,21:PRINT SPACE$(58);:RETURN
640 LOCATE 0,22:PRINT SPACE$(28);:GOTO 590
900 A=USR(0):SUENA=0:DES=0:RETURN
1000 READ A$:IF VAL(A$)<0 THEN ON MEL+1 GOSUB 2000,2010,2020,2030,2040:GOTO 1000
1010 READ B$,C$:PLAY A$,B$,C$:RETURN
1100 P=NOT P:IF P THEN 1200
1110 PUT SPRITE 0,(180,70),15,M2
1120 PUT SPRITE 1,(146,70),15,M1
1130 PUT SPRITE 2,(72,70),15,H2
1140 PUT SPRITE 3,(38,70),15,H1
1150 LOCATE 13,9:PRINT CHR$(220)
1160 LOCATE 13,10:PRINT CHR$(223)
1170 LOCATE 13,11:PRINT CHR$(219)
1180 RETURN
1200 PUT SPRITE 0,(180,70),4,M2
1210 PUT SPRITE 1,(146,70),4,M1
1220 PUT SPRITE 2,(72,70),4,H2
1230 PUT SPRITE 3,(38,70),4,H1
1240 LOCATE 13,9:PRINT CHR$(32)
1250 LOCATE 13,10:PRINT CHR$(32)
1260 LOCATE 13,11:PRINT CHR$(32)
1270 RETURN
2000 RESTORE 3000:RETURN
2010 RESTORE 4000:RETURN
2020 RESTORE 5000:RETURN
2030 RESTORE 6000:RETURN
2040 RESTORE 7000:RETURN
3000 DATA v7o4t200s13m10g4.,v7o3t200s13m10r4.,v7o6t200s13m10r4.
3010 DATA o5g1e8d4c8,c2r8c8o2g2.,r1.
3020 DATA o4a1r8g4.,o3c2r8c8o2g2.,r4a8r4a8g+4a8r4.
3030 DATA o5c1c8d4e8,o3c2r8c8o2g2.,r1.
3040 DATA g1r8o4g4.,o3c2r8c8o2g2.,r4d8e8g8a8o7c8r4r4.
3050 DATA o5c1d8e4g2.e2g4.,o3c2r8c8o2g2.o3c2r8c8o2g4.o3c+4.,r4.c4c8r2.r4.o6g4g8r
2.
3060 DATA g2.e4.d4o4a1.r8,d2r8d8e4.f4d2.d8o2g2.,o6r1.r4a8g+4g8r2.
3070 DATA o5r4g2e8r4d4o4a1.r8,o3d2r8d8o2g2.o3d2r8d8o2g2.,r1.r4d8c+4d8r2.
3080 DATA r4o5g2g4e8d4o4a1r4g4.,o3d2r8d8o2g2.o3d2r8d8o2g2.,r1.r8a16a+16b2a+4.a4g
```

```
8
3090 DATA o5g4g8r2.o4g4.,o3d2r8d8o2g2..r2.f+4g8r4.
3100 DATA o5d4d8r4.e4.o4g4o5c1.c1c4r4.,o3d2r8o2g8a4.b4o3c2.c8o2g2.o3c2r8c8o2g2..
r2.g4.f4e4.c8d4o5a8o6c4.o5a4o6c8e4c8d4o5a8o6c4.r4.
3110 DATA e4c8d4o4a8o5c4.o4a4o5c8e4c8d4o4a8o5c4r8.o3c2r8c8o2g2.o3c2r8c8o2g4r8.e4
c8d4o5a8o6c4.o5a4o6c8e4c8d4o5a8o6c4r8
3120 DATA -1
4000 DATA t120s9m4000o7c16o6b16o7c16d16.t120s9m4000r4.t120s9m4000r4
4010 DATA t120s9m16000c8o6c4e4g8.t120s9m16000o4r8e4g4o5c8.t120s9m16000o4r8c4e4g8
4020 DATA g8f8f4s9m4000f16e16f16g16,o3b8d8o4g2,o3g8b8o4d2
4030 DATA s9m16000f8o5g4o6d4f8.r8o3b4d4o4g8.r8o3g4b4o4d8
4040 DATA f8e8e4o7c8.o6a16,e8g8o5c4r4,c8e8g4r4
4050 DATA s9m8000g8f+8f+8f+8o7c8.o6a16.s9m4000r8f+8f+8f+8r4.s9m8000r8d8a8d8r4
4060 DATA s9m8000g8f+8f+8f+8o7c8.o6a16.s9m4000r8f+8f+8f+8r4.s9m8000r8d8a8d8r4
4070 DATA b8g8e8o7c8s9m32000o6a8..s9m4000g64f+64s9m32000g4r4,d4e4f+4s9m32000g4r4
.g4c4d4s9m32000g4r4
4080 DATA -1
5000 DATA v12t100s9m8000o6e8f8.t100r4.t100r4
5010 DATA g4r4e8f8.s9m8000o5g4e4c4.s9m8000v12o4c4r2
5020 DATA g4r4e8f8.g4e4c4.c4r2
5030 DATA o7c4o6b4a4.g4e4c4.c2r4
5040 DATA a4g4d8e8,d4o4b4g4,v14o3g4r2
5050 DATA f4r4d8e8.o5d4o4b4g4.g4r2
5060 DATA f4r4d8f8.o5d4o4b4g4.g4r2
5070 DATA b8a8g4b4o7c4r4.o5f4e4d4c4r4.g4a4b4v12c4r4
5080 DATA -1
6000 DATA o5t120s9m8000g4.a8g4,o4t120s9m8000g4e4c4.o3t120s9m16000c2.
6010 DATA e4r2.g4e4c4.c2.
6020 DATA g4.a8g4.g4e4c4.c2.
6030 DATA e4r2.g4e4c4.c2.
6040 DATA o6d2d4.d4o3b4g4.o2g2.
6050 DATA o5b4r2.o4d4o3b4g4.g2.
6060 DATA o6c2c4.o4g4e4c4.o3c2.
6070 DATA o5g4r2.g4e4c4.c2.
6080 DATA a2a4.o5c4o4a4f4.f2.
6090 DATA o6c4.o5b8a4.o5c4o4a4f4.f2.
6100 DATA g4.a8g4.g4e4c4.c2.
6110 DATA e4r2.g4e4c4.c2.
6120 DATA a2a4.o5c4o4a4f4.f2.
6130 DATA o6c4.o5b8a4.o5c4o4a4f4.f2.
6140 DATA g4.a8g4,g4e4c4.c2.
6150 DATA e4r2.g4e4c4.c2.
6160 DATA o6d2d8.d4o3b4g4.o2g2.
6170 DATA f4.d8o5b4.o4d4o3b4g4.g2.
6180 DATA o6c4r2.o4g4e4c4.o3c2.
6190 DATA e4r4.g4e4c4.c2.
6200 DATA o6c4.o5g8e4.g4e4c4.c2.
6210 DATA g4.f8d4.d4o3b4g4.o2g2.
6220 DATA c4r2,o4c4e4g4.o3c2.
6230 DATA o6c4r2.o5r2.c2.
6240 DATA -1
7000 DATA o6t80s9m4000g8e8,t80s9m4000r4.t80s9m4000r4
7010 DATA g4g8e8g4g8e8.r1.o4c4r4o3g4r4
7020 DATA o7c4o6g4r4g8e8.r4o5d32.e32.f32.g32.a32.b32.o6c4r4,o4c4o3g4o4c4r4
7030 DATA g4g8e8g4g8f8.r1.c4r4o3g4r4
7040 DATA b4r2g8f8.r4o4a32.b32.o5c32.d32.e32.f32.g4r4.o4d4o3g4o4d4r4
7050 DATA g4g8f8g4g8f8.r1.d4o3r4g4r4
7060 DATA o7d4o6f4r4d8e8.r4o4a32.b32.o5c32.d32.e32.f32.g4r4,o4d4o3g4o4d4r4
7070 DATA f4g8f8e4d8f8e4r2.r1r4o5d32.e32.f32.g32.a32.b32.o6c4.d4r4o3g4r4o4c4o3g4
o4c4
7080 DATA -1
```



Mejorar la rapidez de los programas

Para que nuestros programas se ejecuten más deprisa, existen algunos trucos bastante sencillos. Uno de ellos consiste en utilizar, siempre que nos sea posible, variables enteras. Estas se diferencian de las normales porque terminan con el símbolo %. En el ejemplo se puede apreciar la diferencia de velocidad en dos bucles cuya función es la misma, pero de distinta programación. El primero utiliza variables normales y el segundo, variables enteras (en próximos números explicaremos con más detalle los distintos tipos de variables que existen).

El otro truco consiste en deshabilitar las interrupciones, evitando que el sistema operativo pierda tiempo actualizando el reloj y leyendo el teclado. Esta solución, sólo es útil ejecutando bucles largos que no utilicen la pantalla. Esta función se realiza mediante la instrucción VDP(1) = 208, mientras que para restablecer el funcionamiento utilizaremos la instrucción VDP(1) = 240.

Comprobar varias teclas

En algunos programas, es necesario comprobar si una tecla pulsada está dentro de las necesarias para la realización de cualquier operación. Sin utilizar el método tradicional, vamos a usar una instrucción muy potente del BASIC MSX; INSTR. Con ella, comprobaremos el contenido de una variable en la que habremos guardado las distintas opciones. En el ejemplo, comprobamos si la tecla pulsada es una de las siguientes; N, S, E u O (muy útil en programas de aventuras).

Más rapidez en los programas

```
10 FOR A=1 TO 200
20 OUT 170.0
30 OUT 170.255
40 NEXT A
50 END
```

```
10 FOR A%=1 TO 200
20 OUT 170.0
30 OUT 170.255
40 NEXT A%
50 END
```

```
10 TECLA$="NSEO"
20 PRINT "Pulse N. S. E u O. para continuar"
30 D$=INKEY$: IF D$="" OR INSTR (TECLA$.D$)=0 THEN 20
40 PRINT "Continua la ejecución del programa"
50 END
```

Rótulos móviles

La rutina siguiente nos permitirá mostrar rótulos móviles en cualquier lugar de la pantalla. El mensaje a visualizar se colocará en la variable A$ de la línea 40. Las variables X e Y corresponden a las coordenadas horizontal y vertical, respectivamente. La variable TEMP se utiliza para regular la velocidad de presentación del texto.

Probar a incorporar a vuestro programa y veréis lo original y atractivo que quedan.

```
10 REM rotulos moviles
20 CLEAR 500:KEY OFF
30 X=1:Y=1:TEMP=60
40 A$="ROTULOS MOVILES (C) MSX MAGAZINE 1985......."
50 CLS
60 A$=SPACE$(15)+A$+SPACE$ (15)
70 FOR A=1 TO LEN(A$)
80 LOCATE X,Y,0
90 PRINT MID$(A$,A,15)
100 FOR B=1 TO 100:NEXT B
110 NEXT A
```

Borrar la pantalla

Presentamos una pequeña rutina en código máquina, que produce un borrado de la pantalla de una forma bastante original. Se puede usar en cualquier punto del programa mediante una función USR teniendo la precaución de no utilizarla en SCREEN 0 ó SCREEN 1.

```
10 FOR A=42000! TO 42030!
20 READ X:POKE A,X
30 NEXT A
50 DEFUSR=42000!
60 SCREEN 2:FOR A=1 TO 90 STEP 7
70 CIRCLE(128,95),A
80 NEXT A
90 A=USR(0)
200 DATA 6,3,33,0,0
210 DATA 17,0,24,205,74
220 DATA 0,203,63,203,63,203,63,205,77
230 DATA 0,35,29,32,240,21
240 DATA 32,237,5,32,228,201
```

Rincón del lector

MANEJO DE SPRITES

Desearía que a ser posible me dieran la respuesta a dos preguntas que se me plantean, ya que en los manuales no encuentro información sobre este tema.

¿Cómo puedo seleccionar el choque entre SPRITES?

¿Se pueden hacer sprites de distintos colores?

Antonio Verdi
Alicante

El BASIC MSX no ofrece posibilidad para conocer los números de los SPRITES que colisionan, el método más sencillo es comprobar las coordenadas de cada uno de ellos para ver cuales se encuentran en colisión.

A cualquier SPRITE se le puede poner cualquiera de los 16 colores de que disponen los ordenadores MSX, aunque no es posible definir un SPRITE con dos colores diferentes.

PROBLEMAS CON EL CASSETTE

Desearía preguntarles si un CASSETTE STEREO sirve para grabar y reproducir programas, y cómo se conectarían los cables, ya que tiene dos salidas además del control remoto.

Los cassettes STERO no suelen dar buenos resultados a la hora de utilizarlos con el ordenador, aunque si no hay otro remedio se pueden utilizar teniendo en cuenta los siguientes consejos:

El volumen debe estar entre la mitad y las tres cuartas partes. El control de tono se sitúa en la posición de máximos agudos y se limpiarán las cabezas frecuentemente.

COPIAR PANTALLAS

Hace poco tiempo he comprado una impresora SONY PRN-41C, mi duda es: ¿cómo se pueden pasar pantallas gráficas a la impresora?, también desearía saber si se puede crear más de un intervalo mediante ON INTERVAL GOSUB.

Salvador Maine
Cádiz

Ya que no es posible pasar la pantalla directamente, será necesario construirse un pequeño programa que se encargue de esa misión. Te aconsejamos leerte el manual de la impresora donde se detallan las instrucciones gráficas de la misma, que sin duda te serán de gran utilidad. Respecto a la otra pregunta, te diremos que no es posible tener varias interrupciones de este tipo activadas simultaneamente, aunque si se puede variar la temporización a lo largo del programa.

AMPLIACIONES DE MEMORIA

Voy a comprarme un ordenador de 64 K. Me había decidido ya por un MSX, sobre todo por la posibilidad de ampliar memoria, sin embargo, me ha dicho un amigo que no aumenta la memoria libre para usuario, si esto es cierto, ¿para qué sirven entonces?

Ramón Martínez
Jerez

No es cierto que la memoria libre no aumente, lo que no aumenta es la capacidad de memoria destinada a programas en BASIC. El resto de la memoria puede ser utilizada para almacenar datos, pantallas o programas escritos en código máquina.

DIRECTOR:
Juan Arencibia
COORDINADOR EDITORIAL:
J. Ignacio Rey
REDACCION:
Fernando Garcia, Santiago Gala, Ricardo Garcia, Teresa Aranda.
DISEÑO:
Ricardo Segura y Benito Gil.
Editada por
PUBLINFORMATICA S.A.
PRESIDENTE:
Fernando Bolin.
DIRECTOR EDITORIAL REVISTAS DE USUARIOS:
Juan Arencibia.
PUBLICIDAD:
Emilio Garcia.
Administración:
PUBLINFORMATICA
GERENTE DE CIRCULACION Y VENTAS:
Luis Carrero.
PRODUCCION:
Miguel Onieva.
DIRECTOR DE MARKETING:
Antonio Gonzalez.
SERVICIO AL CLIENTE:
Julia Gonzalez.
Tel. 733 79 69
DIRECCION Y REDACCION:
C/ Bravo Murillo, 377 - 5° A
Tel. 733 74 13
28020 MADRID
PUBLICIDAD Y ADMINISTRACCION:
C/ Bravo Murillo 377 - 3° E
Tel. 733 96 62-96
Publicidad en Madrid:
Fernando Hernando
Publicidad en Barcelona
Lidia Cendros.
C/ Pelayo, 12
Tel. (93) 301 47 00 Ext. 27-28.
08001 BARCELONA.
Deposito Legal: M. 16.755-1985
Impreso en Héroes S.A.
C/ Torrelara, 8. 28016 MADRID.
Distribuye:
S.G.E.L. Avda. Valdelaparra s/n.
Alcobendas (Madrid)
DISTRIBUIDORES:
PORTUGAL: VASP Sociedad de Transportes e Distribuidores. LISBOA
CHILE: CORPRODE SANTIAGO
ARGENTINA: DISTRIBUIDORA INTERCONTINENTAL BUENOS AIRES

SUSCRIPCIONES:
Rogamos dirija toda la correspondencia relacionada con suscripciones a:
MSX
EDISA. Tel. 415 97 12
C/López de Hoyos, 141-5.°
28002 MADRID
(Para todos los pagos reseñar solamente MSX)
Para la compra de ejemplares atrasados diríjanse a la propia editorial
MSX
C/Bravo Murillo, 377-5.° A
Tel. 733 74 13 28020 MADRID

Si deseas colaborar en MSX remite tus artículos o programas a Bravo Murillo 377, 5.° A. 28020 Madrid. Los programas deberán estar grabados en cassette y los artículos mecanografiados.
A efectos de remuneración, se analiza cada colaboración aisladamente, estudiando su complejidad y calidad.

microBYTE PRESENTA... MSX

SENSACIONALES PROGRAMAS EN CARTUCHO Y CASSETTE

FLIGHT PATH 737.



Colócate a los mandos de un jet comercial. Disponemos de control total sobre los mandos del avión, y puedes escoger entre 6 niveles de dificultad.
P.V.P.: CART. 3.490 pts.
CASS. 1.900 pts. 32K.

FRUITY FRANK



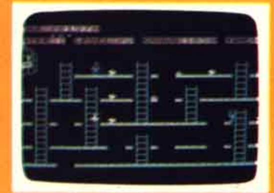
Tu jardín ha sido invádido por monstruos de fruta madura. La única forma de combatirlos es lanzarles fruta fresca del jardín.
P.V.P.: CASS. 1.900 pts. 32K.

SPARTAN X



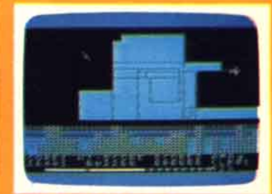
Son muchos los peligros que te acechan. Ten los reflejos bien despiertos, pon tus fuerzas en estado de alerta, y a luchar.
P.V.P.: CASS. 1.900 pts. 32K.

CHUCKIE EGG



Debes recoger los huevos antes de que nazcan los pollitos y se cóman el maiz. Pero ojo con el Pato Loco.
P.V.P.: CASS. 1.900 pts. 32K.

NIGHT FLIGHT



Con tu pequeño avión, debes ir dando luz a la noche, hasta que el cielo esté de nuevo azul. Date prisa en realizar tu misión, de lo contrario...
P.V.P.: CART. 2.900 pts.
CASS. 1.900 pts. 16K

STAR AVENGER



Imagina el juego de batalla más rápido que jamás hayas visto. Piensa además, en los más excitantes gráficos y sus 5 niveles de dificultad. Todo ello es Star Avenger.
P.V.P.: CASS. 1.900 pts. 32K.

GYRO ADVENTURE



Ponte a los mandos de tu helicóptero y combate a los enemigos que se enfrentan a tí. Podrás mover el helicóptero en todas direcciones, mantenerlo en el aire y disparar. **P.V.P.: CART. 2.900 pts.**
CASS. 1.900 pts. 16K.

SUPER CROSS FORCE



Sólo queda una esperanza para la supervivencia ante el ataque de los malvados Morpul. Tú podrás atacarles, con tus naves dispuestas en paralelo o en diagonal.
P.V.P.: CART. 2.900 pts. 16K.

JUMP LAND



Tu mayor obsesión han sido siempre los pasteles, y por ello, te has visto envuelto en situaciones complicadas que has salvado gracias a tus reflejos.
P.V.P. CASS. 1.900 pts. 16K.

ROGER RUBBISH



Los perversos contaminadores de planetas están llenando nuestra galaxia de residuos nucleares. Roger Rubbish es el más famoso recogedor de basuras espaciales.
P.V.P.: CART. 2.900 pts. 16K.

FRUIT PANIC



Un día, Walky, para divertirse se fué al país de los gatos. ¿Cuánta fruta podrá comerse Walky?
P.V.P.: CASS. 2.000 pts. 16K

DIZZY BALLOON



En este mundo hay seres voladores y atacan cuerpo a cuerpo. Si los haces explotar, se irá abriendo el cielo y tendrás la oportunidad de escapar.
P.V.P.: CASS. 2.000 pts. 32K.

CASTLE COMBAT



El castillo galáctico, ha caido bajo la dominación de los Tyrones. Tu nave STAR DUSTER, está preparada para el combate. ¿Te atreves?
P.V.P.: CART. 2.900 pts. 16K.

NICK NEAKER



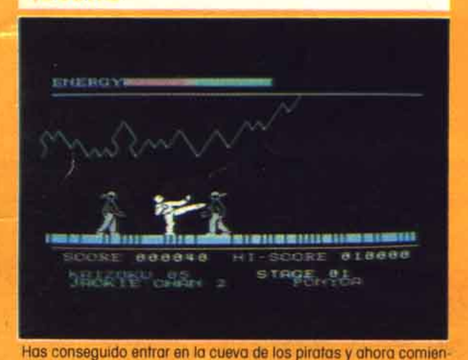
Cuando estás dormido, muchas cosas suceden a tu alrededor. Algunos objetos de tu casa toman vida, como en el caso de la zapatilla NICK.
P.V.P.: CART. 2.900 pts.
CASS. 1.900 pts. 16K.

CHAMP



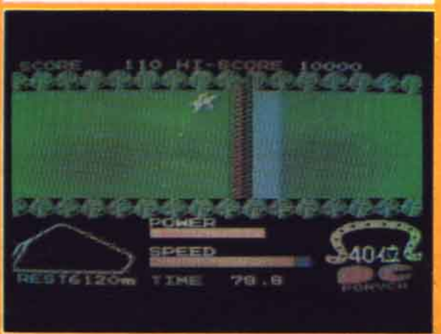
Champ es un completo Ensamblador/Monitor para tu MSX. Champ te permite escribir y trazar programas en código máquina con el mínimo esfuerzo.
P.V.P.: CART. 3.890 pts.
CASS. 2.400 pts. 32K.

KARATE



Has conseguido entrar en la cueva de los piratas y ahora comienzan tus problemas. Los murciélagos gigantes, moradores de esras cuevas pueden chuparte la sangre. Cuando te encuentres con los piratas, deberás enfrentarte a ellos con tu depurado estilo de Kárate. **P.V.P.: CART. 3.490 pts. CASS. 1.900 pts. 32K.**

GRAND NATIONAL



Si te gustan las carreras de caballos, no te quedes como un espectador, participa. Ahora puedes correr con tu caballo, en la más prestigiosa carrera del mundo, el GRAND NATIONAL.
P.V.P.: CASS. 2.000 pts. 32K.

ENVIENOS A **MICROBYTE**
P.º Castellana, 179, 1.º - 28046 Madrid

Nombre
Apellidos
Dirección
Población
D.P. Teléfono

ENVIOS GRATIS

JUEGO	Cart.	Cass.	Precio	TOTAL

PRECIO TOTAL PESETAS

Incluyo talón nominativo ☐
Contra-Reembolso ☐

Pedidos por teléfono **91 - 442 54 33 / 44**